INSTITUTE FOR SIMULATION AND TRAINING

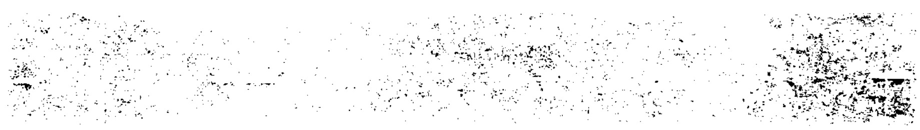
[illegible]
[illegible]
[illegible]

[illegible]

Summary Report

The Third Workshop on Standards for the Interoperability of Defense Simulations

Volume I: *Minutes from Plenary Session and Attendees List*



IST

[illegible]
[illegible]
[illegible]

University of Central Florida
Division of Sponsored Research

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION unclassified	1b RESTRICTIVE MARKINGS None
2a. SECURITY CLASSIFICATION AUTHORITY N/A	3 DISTRIBUTION/AVAILABILITY OF REPORT Approved for Public Release; Distribution is Unlimited
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE N/A	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) IST-CR-90-13	5 MONITORING ORGANIZATION REPORT NUMBER(S) IST-CR-90-13
6a. NAME OF PERFORMING ORGANIZATION Institute for Simulation and Training 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION Project Manager for Training Devices
6c. ADDRESS (City, State, and ZIP Code) 12424 Research Parkway, Suite 300 Orlando, FL 32826	7b ADDRESS (City, State, and ZIP Code) 12350 Research Parkway Orlando, FL 32826
8a. NAME OF FUNDING/SPONSORING ORGANIZATION DARPA/TTO 8b. OFFICE SYMBOL (If applicable)	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER N61339-89-C-0043
8c. ADDRESS (City, State, and ZIP Code) 1400 Wilson Blvd. Arlington, VA 22209	10 SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT ACCESSION NO.

11. TITLE (Include Security Classification)
Summary Report: The Third Workshop on Standards for the Interoperability of Defense Simulations Volume I: Minutes from Plenary Session and Attendees Lis

12. PERSONAL AUTHOR(S)
(Edited by) McDonald, Bruce; Pinon, Christina

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Summary	FROM ____ TO ____	1990, August 7-8	167

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Communication Protocols, Terrain Databases, Performance Measures

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

This report presents a summary of the activities of The Third Workshop on Standards for the Interoperability of Defense Simulations sponsored by DARPA and PM TRADE, and hosted by IST/UCF on August 7-8, 1990.

The primary goal of this workshop was to recommend revisions to the proposed Draft Standard for Protocol Data Units in Distributed Interactive Simulation (DIS) published in June 1990 by IST.

This volume contains the minutes of the plenary session and a list of attendees.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21 ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED
22a. NAME OF RESPONSIBLE INDIVIDUAL Mike Garnsey	22b TELEPHONE (Include Area Code) 407-380-4816 22c. OFFICE SYMBOL AMC PM-TND-EN

DD Form 1473, JUN 86 Previous editions are obsolete. SECURITY CLASSIFICATION OF THIS PAGE

INSTITUTE FOR SIMULATION AND TRAINING

Contract Number N61339-89-C-0043
PM TRADE
DARPA

August 7-8, 1990

Summary Report

The Third Workshop on Standards for the Interoperability of Defense Simulations

Volume I: *Minutes from Plenary Session and Attendees List*

Editors:
Bruce McDonald
Christina Pinon

This report is informational and does not express the opinions of PM TRADE or DARPA

Institute for Simulation and Training
12424 Research Parkway, Suite 300
Orlando FL 32826

University of Central Florida
Division of Sponsored Research

IST-CR-90-13

PREFACE

Purpose

The purpose of this report is to present the minutes from the Third Workshop on Standards for the Interoperability of Defense Simulations. This workshop took place in Orlando, Florida on August 7-8, 1990 and was hosted by the Institute for Simulation and Training (IST), a part of the Division of Sponsored Research for the University of Central Florida (UCF).

This continuing work on standards is sponsored by the Defense Advanced Research Projects Agency (DARPA) and is administered by the Project Manager for Training Devices (PMTRADE).

Background

This is the third workshop concerning the development of technical standards for networking defense simulations. These standards are intended to meet the needs of large scale simulated engagement systems which are being used increasingly to support system acquisition, test and evaluation, tactical warfare simulation and training in DoD. The primary goal of this workshop was to recommend revisions to the proposed **Draft Standard for Protocol Data Units in Distributed Interactive Simulation** (DIS) published in June 1990 by IST. Another goal of the workshop was to continue work towards developing standards in other areas of Distributed Simulation.

Workshop Summary

The two day workshop focused on three major topic areas. These are: Communication Protocols, Terrain Databases, and a new area called Performance Measures.

Discussions in the Communication Protocols Working Group were led by Joe Brann, IBM and Mike McGaugh, McDonnell Douglas. This group concerned itself with resolving issues related to the Draft

Standard. Recommendations were made for incorporation in the revised draft standard which will be published in January 1991. One subgroup, the Communications Architecture Subgroup, met separately. This group focused on issues related to communications architecture. In particular, this group sought to more clearly define the services that a DIS requires from the communication architecture supporting the DIS application. This group was led by Steve Blumenthal, BBN and Al Kerecman, USACECOM.

Discussion in the Terrain Database Working Group was led by Mr. Dexter Fletcher, IDA. This group continued its work with representation and interpretation of terrain data.

A new working group, the Performance Measures Working Group, met to discuss human and equipment performance measures. This working group was led by Dr. Bruce McDonald, IST. This group focused on identifying information within a DIS that is considered essential for accomplishing necessary training objectives.

This report has been issued in three volumes. Volume I contains the minutes for the plenary session and a list of attendees. Volume II contains the view-graphs from the plenary sessions. Volume III contains the view-graphs used in presentations made in the individual working groups.

CONTENTS - VOLUME I: Minutes and Attendees List

CONTENTS - Vol. I (cont)

CONTENTS - Vol. I (cont)

CONTENTS - VOLUME II: View-Graphs From Plenary Sessions

CONTENTS - VOLUME III: View-Graphs from Working Group Presentations

Acronyms used in this document. The acronyms used in this document are defined as follows:

ABS	-	Advanced Battle Simulation
ASD	-	Air Force Aeronautical Systems Division
ADST	-	Advanced Distributed Simulation Technology
ANSI	-	American National Standards Institute
ASCII	-	American Standard Code for Information Interchange
ASW	-	Anti-Submarine Warfare
BAM	-	Binary Angle Measurement
BFIT	-	Battle Force In-port Trainer
CIG	-	Computer Image Generator
CNO	-	Chief of Naval Operations
CNA	-	Center For Naval Analysis
CWI	-	Continuous Wave Illuminations
C3	-	Command, Control and Communications
DARPA	-	Defense Advanced Research Projects Agency
DIS	-	Distributed Interactive Simulation
DRN	-	Data Representation Notation
DSNT	-	Distributed Simulation Networking Technology
DWS	-	Distributed Warfare Simulation
ECM	-	Electronic Countermeasures
EHF	-	Extremely High Frequency
ENWGS	-	Enhanced Naval Warfare Gaming System
EO	-	Electro-Optical
ESM	-	Electronic Warfare Support Measures
EW	-	Electronic Warfare
FDDI	-	Fiber Data Distributed Interface

GOSIP	-	Government Open Systems Interconnection Profile
GRWSIM	-	Ground Warfare Simulation
HF	-	High Frequency
HRL	-	Air Force Human Resources Laboratory
ICAF	-	Industrial College of the Armed Forces
IEEE	-	Institute of Electrical and Electronic Engineers
IFF	-	Identify Friend of Foe
IR	-	Infrared
IST	-	Institute for Simulation And Training
ITD	-	Interim Terrain Database
ISO	-	International Organization for Standardization
ISODE	-	ISO Development Environment
JESS	-	Joint Exercise Simulation System
JTIDS	-	Joint Tactical Information Distribution System
LAN	-	Local Area Network
LF	-	Low Frequency
MIPS	-	Millions of Instructions Per Second
NAVAIR	-	Naval Air Systems Command
NAVSEA	-	Naval Sea Systems Command
NIST	-	National Institute for Standards and Technology
NOSC	-	Naval Ocean Systems Center
NTSC	-	Naval Training Systems Center
OPNAV	-	CNO Office
OSD	-	Office of Secretary of Defense
OSI	-	Open Systems Interconnection

PDU	-	Protocol Data Unit
PM TRADE	-	Army Project Manager for Training Devices
PRF	-	Pulse Repetition Frequency
RISC	-	Reduced Instruction Set Computer
SAFOR	-	Semi-Automated Forces
SAM	-	Surface To Air Missile
SINCGARS	-	Single Channel Ground and Air Radio System
SHF	-	Super High Frequency
SIMAN	-	Simulation Management Protocol
SIMNET	-	Simulator Network
SIMSPO	-	Air Force Simulator Systems Program Office
SPAWAR	-	Naval Space and Warfare Systems Command
TCP/IP	-	Transmission Control Protocol/Internet Protocol
UHF	-	Ultra High Frequency
USACECOM	-	Army Communications and Electronics Command
USAETL	-	Army Engineering Topographic Laboratory
UTSS	-	Universal Threat System For Simulators
VLF	-	Very Low Frequency
WAN	-	Wide-Area Network
WAREX	-	War Fighting Exercise
XTP	-	Express Transfer Protocol

Welcome, Announcements and Review of Agenda
Brian Goldiez

Good Morning! I'm Brian Goldiez from the Institute for Simulation and Training at the University of Central Florida, and I want to welcome all of you to the Third Workshop on Standards for the Interoperability of Defense Simulations.

For those of you who don't know, the Institute's been involved in networking research on simulators for over three years. Our work involves a wide span of research from basic research involved with the fundamentals of networking technology to applications of existing networking methods to simulation's unique problems.

When the Institute hosted our first meeting about a year ago we were searching for some common ground to develop a standard to connect simulators and simulations. The SIMNET model proved to be a good place to start, and many modifications have been suggested and incorporated into the draft standard that you all have. We have also uncovered some areas which require additional work to have a completely rounded out standard. A few of these areas include testing, and the correlation of images and topics that transcend the visual world. It goes into the other portions of the electromagnetic spectrum. Other areas requiring additional investigations will be discussed over the next two days, and we look forward to your recommendations and comments on the standard, and on additional areas that you feel need to be looked at in the future.

Suffice it to the say, though, that the draft standard we have and that we will be discussing over the next several months is a draft and it's changeable. It's also expandable in terms of scope, but we are getting closure. The Institute is taking on the challenge of developing a military standard in the relatively short period of time of about one year. In order to accomplish this task, we've solicited inputs from interested parties. You all have responded with your inputs and we appreciate that. We received

over fifty (50) position papers for the January meeting and those papers, we feel, have been dispositioned in the standard and the rationale document which you now have. For this workshop, we received thirteen (13) position papers. I can only say that I think this downward trend in the number of received position papers shows that we are coming to grips with the problems and are resolving them on a case by case basis.

This is a simple chart for those of you who are new. We have a steering committee to create the standard and we've divided the work into three parts. One part deals with what we call terrain data bases. I think that group, you'll find in the next day or so, is going to expand in scope outside of terrain. Terrain Databases are headed by Mr. George Lukes of the Army Engineering Topographic Labs. For these meetings, in the next two days, Dr. Dexter Fletcher will act in George's place. Dexter is from IDA. Communications Protocols is headed by Dr. Ron Hofer from PM TRADE. There's a new group, that we put together as a result of the meeting in January, that deals with performance measures and is headed by Dr. Bruce McDonald of IST. Underneath each of these groups are sub-groups that deal with specific issues. As time has gone on, these sub-groups have kind of ebbed and flowed in terms of both productivity and merging of groups together, so it's rather fluid underneath. It's really the issues that you all point out that determine the complexion of these sub-groups.

I'd like to express some "thank you's" at this point to the Steering Committee, and to identify them! Dr. Bruce McDonald from the Institute, is really the guy who spearheaded the work that went into the document that you all have. Other members of the Steering Committee are Ron Hofer from PM TRADE, Gene Wiehagen from PM TRADE, LtCol Jimmy Shiflett, Major Jim Wargo and LCDR Dennis McBride from DARPA, George Lukes and LtCol Steve Sarner from SIMSPO, Bill Harris from the Naval Training Systems Center, Neale Cosby and Dexter Fletcher from IDA. Our industry participants: Dr. Joe Brann

from IBM, Sam Knight from CAE-Link, Dr. Duncan Miller from BBN and Richard Moon from E&S. Tom Nelson was in the group but he's been replaced by Mike McGaugh of McDonnell Douglas.

We couldn't have done anything without these guys and their guidance and their leadership through this whole effort. The way the process worked was that when we got position papers, we were looking for the direction that the standard needed to take. It was not a unilateral decision on the Institute's part. The Steering Committee deals with issues, and if assignment of work or responsibility has to occur, they make those assignments in a democratic fashion. I don't want to forget the people from IST. There was a group of three people who spent a lot of extra time putting these documents together, studying the issues, studying the work we were doing at the Institute, and putting together the document that you see. I'd like to identify them. They will be up on the podium in a little while. One is Chris Pinon, the other is Bob Glasgow, and the third is Karen Danisas. Without those three people there wouldn't have been a document at all. We also had a lot of support from students and our clerical staff to put four hundred (400) copies of a document together.

As I said, we received thirteen (13) position papers by the cutoff date, which was about two weeks ago, and the Steering Committee selected nine of those for presentations. Personally, I want to thank all thirteen of the people who submitted position papers. I want to assure all thirteen of you, or your companies, that all of those papers will be considered in the standard and in the revision to it. It was just that of the topics were really not appropriate for a large audience. But all of those position papers, I might add, are available to anybody who wants them.

I'd like to introduce our speaker, the Honorable Christopher Jehn. Mr. Jehn was sworn in as the Assistant Secretary of Defense for Force Management and Personnel, on November 20, 1989. After

serving on the faculties of the George Washington University and the University of Illinois, Chicago Circle, Mr. Jehn joined the research staff of the Center for Naval Analysis in 1972. He served at CNA in various capacities and was Vice President, Navy Marine

Corp. Planning and Manpower Division before joining the Bush Administration. Mr. Jehn was born and raised in Chicago and currently resides in Virginia with his wife and daughter. He holds a B.A. from Belloit College and an M.A. in Economics from the University of Chicago. Mr. Jehn.

Keynote Address, The Honorable Christopher Jehn

Thank you Brian. It's a privilege to talk to all of you today. There are few more timely and important topics than that of Standardization of Simulations and I am going to try and tell you why I think so.

In the year since your last conference the world has undergone momentous change. The Berlin Wall has come down, checkpoint Charlie is gone, Germany races toward a unified future, the Soviet Union looks inward, faced with serious internal problems, and citizens of East Bloc nations look to new friends in the West as they taste freedom their first time in more than forty years.

The men and women in the Armed Forces must take enormous satisfaction in knowing that their efforts these past forty years have been rewarded. But now they face significant change and new challenges.

The President's Defense Budget for fiscal 1991, proposes to reduce real spending two percent (2%) per year over the next five years. Congress, of course, is proposing much bigger reductions. In any event, we know that we will simply have less money for defense and that will have numerous consequences.

Obviously it will affect the development and fielding of new weapons systems and the modernization of systems already fielded. The pace will be slower, more selective, and will reflect new thinking about the threats we face.

Our forces overseas will be scaled back, consolidated, and many overseas facilities may be closed. And the same is true of course, for stateside facilities. There will be fewer people in uniform and to keep the best and brightest we must provide them with challenging work and innovative training.

But operations and maintenance, so key to training and readiness, are threatened too, by both resource reductions and environmental restrictions. Many in Congress said they share our intention to protect training but we shall see. But to strengthen their resolve I think we must increase our use of emerging technologies that will allow more practice for the same or less cost, thereby reducing O&M costs.

There will be less opportunity to let large teams go to a range to fight force-on-force battles, fewer large exercises where joint forces can practice fighting together, and fewer command and staff exercises that are essential to learning how to integrate all the combatants and support units in a big theatre. Today, we do not have enough opportunities to allow commanders and their teams to achieve mastery of war fighting skills, and in the future, there should be even fewer opportunities. A key difference between amateurs and experts, though, is practice, practice, more practice. So we must create ways and means of getting enough practice in today's environment.

This is particularly true for our Reserves who have less time for training. Their responsibilities may increase in today's changing world but their situation isn't going to change. They will remain widely dispersed across the country, holding other full-time jobs, dedicating only weekends and time in the summer for training.

That many of our fundamental military strategies and doctrine are under review, victims of the changing world, further complicates these problems. We will continue to expect our military forces to deploy rapidly and fight effectively, of course. But we need to train them more effectively and cheaply, and we also need better tools to assess the new strategy and doctrine we'll develop for the changing world.

All of this means we've got to train, buy, and manage smarter than before. And that brings me to the importance of the work you are doing here.

There is not a senior leader in the Administration, Congress, or the Military who hasn't recognized the central role that simulation must play in the future defense of our nation, as well as that of our Allies. Here, I am using the broadest definition of the term, **simulation** to mean conventional simulators, wargames, computer-assisted exercises, and all other exercises. Indeed, even the use of actual weapons systems in training, since all training, short of war, is really simulation. So I'm talking about the full range of synthetic battlefield environments which we use to train, practice, and master warfighting skills, as well as to evaluate new weapons systems ideas in full battle context.

I use this inclusive definition purposely because I believe that just as the major development and simulation technology of the last five years was simulator networking, the next important development will be the extension of interoperable simulations to all types of training.

Now why is it important to be able to interconnect simulators and simulations? I bet most of the people in this room know, but let me just bore you with an answer. For years, we've been buying a variety of simulators; some which have been very capable, many of which are expensive, and most of which cannot talk to one another. But if our simulators cannot talk to one another then we can only use them to teach individual and small unit skills. Further, if they are so expensive that we can only afford a few, then their total contribution will be limited. So I'm drawn to the conclusion that the day of the stand-alone, multi-million dollar simulator may be over.

Simulators must be networked to support team and unit missions so we should require simulators that can talk to each other using standard communications protocols and compatible digital databases. Given the importance of well-orchestrated, coordinated joint operations, we need simulators from all the Services to interoperate with one another in joint fashion. Thus, the importance of the work you are performing here to set the standards for simulator interoperability. It's critical for the years ahead.

But interoperability should be just the beginning. A national simulation capability that is carried on a world wide communication network (we sometimes refer to this as the Defense Simulations Internet) is the next logical step. This has been called, **Seamless Simulation.**

And why is this important? First, of course, it might save us some money. Within DoD there are scores of simulations in existence or under development. They range from battalion and brigade simulations that exercise the unit staff through theatre simulations that exercise senior NATO commanders and staffs in Europe. Few, if any of these simulations can talk to one another, just like the older simulations I mentioned a moment ago. There are no common architectures, no compatible databases, and no easy way to share parts of one simulation with one another. Each is its own development and each costs a great deal of money. Requiring interoperability could permit the development of modular simulations building on previous work at lower cost. More important, interoperability could lead to better simulations at all levels. We could systematically grow the battlefield to serve various needs, use lower-level simulations to help improve the behavior of simulations representing higher echelons, and use upper-echelon simulations to provide the backdrop for lower-echelon warfighting.

But we've got a lot of important and difficult work to do before all this.

Your work and agreeing upon DoD standards, which I hope will be suitable as international standards, needs to continue moving swiftly. Pioneering work is already taking place in DARPA, and the research networks that DoD operates on a daily basis have provided good data on the robustness of the current set of protocols. Don't throw those lessons away -- adapt and use them. But be careful that your standards are flexible enough to serve new technology and changing circumstances. Too often, standards become barriers to progress rather than facilitating it.

We need to better understand how databases for battlefield sensors like RADARS and for intelligence systems can be made compatible with the other combat and support systems in battle. The problem is challenging, but solvable.

We need to develop architectures for seamless simulation. There has to be coordinated DoD program for developing, prototyping, and building the truly joint-simulation environment for the future. We can no longer take a piecemeal approach.

To that end, I will take the lead in OSD for training. With Charlie Herzfeld, the Director of Defense Research & Engineering, I'll develop and manage an integrated approach to simulation.

The first step is your work here, which is DoD standards. We're counting on you.

An integrated, coordinated development program and effective policy oversight could take several forms but I think these are some of the key elements it would have:

First, an executive steering committee consisting of assistant secretaries or the equivalents, to provide department-wide guidance. Dr. Herzfeld and I just talked very recently, within the last week or so, about doing just that.

A small shop in my staff to support this committee in the training area.

A joint-program management office focusing on standards, integration of models, coordination of CINC's, and other requirements and facilitation of exercise execution.

DARPA probably needs to have a Simulation Technology Office to aggressively manage the R&D, both horizontally and vertically.

Over the next several months my staff and I will promote a plan of this sort in conjunction with Charlie Herzfeld's staff. It will be a plan to assure effective, affordable simulation-based training for the 21st century. And of course, Defense Research & Engineering sees simulation as offering great promise in the weapons development program as well.

I hope we see DoD with a world-wide, dial-up conference call, joint training system not unlike the global telephone system you can plug in anywhere and train anywhere. Imagine, why leave dispersed personnel dialing in to fight a credible enemy without leaving war rooms, ready rooms, or National Guard Armorys, or even their own kitchen tables? Conference-call training.

To realize that vision though is going to require a concentrated effort and a lot of effort from all of us. So for those of you from industry, please carry back to your corporate managers my personal thanks for their resource commitment to this important endeavor. To you in government, also please convey to your leadership my appreciation for their dedication to the

establishment and maintenance of defense-wide simulation standards for joint-interoperability. To all of you, I must say that our future military forces will be better because of the good work you are going to be doing here. And a special thanks to the Institute for Simulation and Training at the University of Central Florida for hosting this important workshop. I look forward to seeing those DoD Simulation Standards in December of this year. Thank you very much.

(Question) Do you see the DoD having to channel or direct funding in support of the SIMNET technology? And have the Services come up with training requirements? Do you see that as maybe getting in better shape than some other things?

Well, rather than respond in terms of funding for SIMNET technology let me just say what I said in the brief talk. Dr. Herzfeld and I certainly believe simulation technology is an important wave of the future. Not just in training, it's important there (and that's my concern of course), but Dr. Herzfeld sees it as sort of playing a key role in the weapons development process, and I think I agree with him on that score too. So, the two of us together have agreed that we've got to push simulation technology. That has been my view and I think he shares this view. I don't want to put words in his mouth. Certainly, in my view it has been a diverse, unfocused effort that has had a lot of notable accomplishments, but I think there hasn't been this clear vision of where we are going and what we want to do with the progress when we are finished. And providing that vision, and also the stimulus to the Services and other agencies to pour money into this area as appropriate, is exactly what Dr. Herzfeld and I plan on doing. So to the extent I can speak for myself and Charlie Herzfeld I think I can say, "yes". The answer to your question is yes. We want to push here. This is an area, as far as I am concerned, whose time has come. And the real question isn't what to do, but really, how to do it.

Well I thank you all again. And once again good luck in your efforts here and I wish you very well. It's important work. Thank you.

Standards Progress, Dr. Bruce McDonald, IST

I want to tell you about the progress on Standards for Interoperability of Defense Simulations. The funding came from DARPA and the program is administered by PM TRADE. Before we get into what the standards progress consists of let's consider where we are headed.

Now, what is Distributed Interactive Simulation (DIS)? That's the term we are using for this program -- DIS. DIS is intended for combined arms exercises where the emphasis is on team interaction. That is what its primary application is. It is not intended for initial operator skills-training. That is a more detailed area. Generally, we assume that the person who goes into one of these exercises has already learned to do his task as an individual. The primary emphasis of this standard is on interacting as a team member with the overall group.

DIS will be used primarily to train teams to function smoothly with other teams. DIS is also used to evaluate the ability of equipment to work smoothly with other equipment and personnel. The intention of this standard is to help a team interact before they go to war. The Standard will also help evaluate equipment that is under consideration or is in development.

Now the applicability of the standard is to communications between dissimilar simulations. If you have two devices, one built by one company and another built by another company, this standard is designed to help those two simulators work together. The Standard will also apply to anything over long haul networks, and between simulations and actual equipment. Suppose, for example, you've got a wargame and you wish to interact with the equipment. Also suppose that you've got some tactical computers that you want to introduce in the future. We envision the tactical computer as

being able to communicate directly with the simulation using this protocol.

As an example of what we mean here, say that you already have some simulators working on a local area network right now. We do not envision having to go back and completely redo the protocol. If they are already working together now there is no reason to change the protocol. But if you've got this simulator working with simulators there, or going to Europe, then you would need this protocol to go over the long lines. Or, if you already have a group of simulators working together and you bring in a dissimilar device into the local area network, then you would want to use this new protocol for that new device to communicate with the local area network. So basically we are not saying go back and redo all of the things that are now talking to each other. But if you want to interact outside this one area then you would need to use this protocol. That is the emphasis.

Okay, what is the history? The first workshop was in August, 1989. The second workshop was in January, 1990. That is where we agreed to write the draft standard and had our first meeting on it. We had sub-group meetings in March to define the standard more and then we had the initial draft standard which came out in June. This is the one which you have received.

What was the scope? What was IST asked to do? We were to standardize Distributed Interactive Simulation at the Protocol Data Unit level. Our goal was not to define an architecture or to define what goes on inside the simulations. The standard objective is at the Protocol Data Unit level. Chris will discuss that in more detail in her presentation.

We wrote the standard to standardize the PDUs. I have worked on a standard before and the format of a standard will not let you say: "why?". It just says: "Do this.". And so in order for the

standard to make sense you have to write a rationale document. So we prepared a rationale document to explain why we did each thing. We went through and said this sub-committee recommended this, this position paper said this. So the rationale document backs up why the standard says what it says. And also, a large number of issues came up other than the PDUs. And so we documented those in the rationale document. That was the approach. Basically, the rationale document covers "why" and other issues. As things come up and are recommended, we will summarize them in the rationale document and if there is enough consensus, then they will go into the standard. That is sort of the process.

The approach for the standard was to aggressively propose workable solutions. One approach is to simply sit back and as the sub-committees recommend things, we do them. We decided that if we did that we'd be here five years from now. So we took an aggressive approach. We didn't wait for the perfect solution and we very much avoided the "s" word -- needs further study. If we thought it would work we put it in there. A good example of that is the Emitter PDU. The sub-committee only recommended a Radiate PDU for RADAR that used a data base to define parameters. We needed something to handle the signal intelligence/electronic warfare area and it probably should be database oriented. With that information, I dreamed up the Emitter PDU. I'm a member of the Association of Old Crows and the Armed Forces Communications and Electronics Association so I figured I could take a shot at it. I knew it wouldn't be perfect. I knew that I would get some criticisms, but I did accomplish my goal. I got a lot of wallflowers back there off the wall and I'm getting all kinds of comments. Now if I had said, "this deserves further study," a year from now I'd be saying, "this deserves further study". So, I think I forced the issue, and I've gotten some very good comments. I think with the comments we can produce a very good Emitter PDU. The other thing that we've done is we've allowed for expansion and modification. We went ahead and took the position that there is

always room for additions and modifications. So that is the philosophy that you will see behind the rationale document and the standard.

This brings up communication protocol. We were told we didn't have to cover that, but when you get into coming up with these PDUs you start looking at what kind of communications protocol is there. There is the Open Systems Interconnection protocol. The National Institute for Standards and Technology issued a Federal Information Processing Standard two years ago which says that all communication systems must include government OSI profile, which is called GOSIP, as of next week. This applies to all new systems and all major upgrades. This applies to communications systems, not to training devices, so we don't feel that this standard has to comply with OSI. But you don't have to be too smart to figure out that if you want to go with a protocol, you go with the one that is coming up, not the one that is going out. So TCP/IP will probably fall back and OSI will take over. So it is obvious we should comply with OSI. We are assuming in our standard that this communications protocol will comply with OSI.

OSI was developed by the International Standards Organization. It is meant for communication from one computer to another where you have hand shaking, the whole bit. You really can't do real-time with that full blown OSI. So what we are looking at is getting a higher performance version of OSI. So IST has started a study of OSI protocol. We are going to measure the performance of the current TCP/IP protocol which is dominate now. We are going to create OSI stacks. OSI is a seven-layer stack. Chris is going to talk about that. We are going to measure performance. We are using the ISODE environment to develop these stacks and we are going to use the lightweight presentation protocol. That is sort of a turbo version of OSI. We will be using TCP/IP, Express Transfer protocol and the other various protocols. And we will be using this on ETHERNET, Token Ring, and Fiber Optics. And so the

idea is how do we get enough performance out of OSI for real-time? Fortunately, I got into the old boy network and found out that SPAWAR has come up with a dual OSI stack. One part of the stack is full blown OSI, you know, "are you listening, here's the data, did you get it?", the full stuff. There is a second stack that is stripped down and much faster for tactical communication data -- real-time data. And so we tracked it down to an engineer in Dahlgren, Virginia. We have got copies of that software and we are going to see if that will solve our problem. We think that will be fast enough. So we are not going to charge off and dream up something new. We checked around and found out that the Navy has already developed something we think will work. We are going to see if we can get that to take care of our communications requirements.

Why are you here? How do you impact the standard? The number one way is to: contribute to sub-committee recommendations. Tomorrow you will break up into sub-groups and go to which ever one you feel you can contribute the most to. At the end of the day your Chairman will summarize what has been decided in your groups and those will be the strongest recommendations to us to include in the standard. The fastest way to impact this standard is to speak up in the sub-committee meetings. The other way is to submit position papers. We mentioned earlier we received fifty position papers and we pay an awful lot of attention to them. A lot of the decisions we made came directly out of those position papers. If you have some ideas and you didn't articulate them well enough in the sub-committee meetings, then submit a position paper. Don't just come up and say, "you need to do so and so". Write it down.

Okay. Where do we go from here? The main reason we are here is to finalize the current set of PDUs. We have this batch of protocol data units that are in the standard, and we want final closure on those PDUs, and let's get that put to bed. That's number one.

Then you can recommend other PDUs. We need PDUs for simulation management, performance measures, and data collection. We need input on that, and if there is enough closure we may be able to include it in the standard for the next pass. If not, then we will put it in the rationale document for consideration and it can go in there later. And we need to develop an overall application description. We really don't have a full description of how the architecture will go together yet. So that is another thing we need to address and we will put that in the rationale document.

The next thing we need to do is develop or select standards for the terrain database or emitter database. We have had a lot of discussion on that, we need to get closure and we need to figure out what kinds of emitter databases we need in there. The signal world is so rich I just can't see transmitting all of those parameters over the lines. We are going to have to have databases in every simulator, I think. So we need to select those databases and define how we are going to interact with them.

In the workshop following this workshop, one of the issues is UTSS -- (Universal Threat System for Simulators). We are assuming that will serve some of our purposes. We need to develop recommendations for level of fidelity. That is a very important issue in this process because if you get carried away on level of fidelity you are going to price this thing out of the market. It is going to get so expensive that you can't do it. And so we need to address level of fidelity.

We need to address security. You can use fake parameters over the lines to keep it unclassified but when you start doing tactics that is going to make it classified very quickly. So I am a strong believer in the fact that this thing is going to have to be classified "SECRET". And so, we are going to need to encrypt it. We need to do some studies to see if the commercially available

encryption programs are fast enough to meet our requirements. And then we need to look at unmanned forces. In order to keep the cost of this system down we have to have semi-automated forces to serve as the bad guys or as extra good guys. In order to keep the system cost down, we need to look more and more at unmanned forces. One of the requirements: How do you talk to them? What kind of behavior is required? And I am hoping we have tactical people in the audience. We definitely need some more guidance from tactically experienced people.

What happens after the workshop? The position paper deadline was 1 November. We had a meeting last night and we are going to move that date up to 1 October because we want to get this thing together a little faster for review by the Steering Committee. So if you come up with an idea after this workshop and you want to submit something we want it in here at IST by 1 October. Okay?

We are going to complete the draft standard. Originally we were saying the end of December. I don't really want to be there till midnight December 31st so I think I am going to promise probably the first or second week in January. So we can meet that deadline without any problem.

You can obtain copies from UCF the same way you got the standard last time. If you wanted it this time and you weren't at the last workshop you had to purchase one. It will be up to our sponsor whether we send them out to you for free or whether it will cost you twenty bucks. The next workshop will probably be in March. That will give you time to look things over and react. The idea is that, hopefully, in March we'll have put to bed the PDUs in the standard. The next workshop will go deeper into terrain databases, fidelity and performance measures.

What is the approval process? Once IST delivers this standard, then what? PM TRADE is going to be the executive agent.

They will get a project number from Battle Creek, Michigan then get approval by PM TRADE, NTSC, and the SIMSPO. At this stage, what we want is for the three commanders to have their committees evaluate the standard and say, "Is this enough to purchase simulators with?" If the Standard looks good to them they will sign it and it can then be used for training devices. Then we'll coordinate with industry and DoD, resolve any of the comments and approve the documents. And at that point PM TRADE will be responsible, as the executive agent, for maintaining the documents. Then you send it to Philadelphia so they are available like any other standard you're familiar with. Then, PM TRADE and ANSI will convert the standard into an ISO standard. Right now it's sort of oriented toward our system. And we do want this to eventually be an International Standards Organization standard so that the British, Germans, French, or anybody can interact with us. PM TRADE will work with ISO, to obtain approval as an ISO standard. Once it becomes an ISO standard we no longer need the MIL standard. So we'll cancel that.

(Question) Where do you bring in the equipment evaluation personnel?

Hopefully they're here.

(Question) No, I mean in the process. Where do you bring in the simulations used in equipment evaluation?

Once we start the process of developing a DoD Standard, we are getting into all simulations, not just training devices. We had an interim step because we have training devices coming to procurement. We need something going very quickly. From there (Coordination With DoD and Industry) on we're bringing in, (hopefully we've already got the input from the wargame people at this time and this should flow right on up), but from here on up it covers everything. And then PM TRADE as executive agent would be

responsible for maintaining currency of the standard from that point on.

Okay, I'd like to preach a little bit. When specifying accuracy requirements remember to differentiate between internal models of your own platform, which requires high accuracy, no question about that, and representation of other platforms which requires less accuracy. Some of the comments that we've gotten on the standard are that, "but the system can resolve so-and-so milliradians". I know that the real system can do that but the PDUs are telling another system where that thing is aimed. And so, I don't need to know that the gun is aimed within 1 milliradian when I can only see it as a human being within 3°. So please keep that in mind when you start specifying resolution. What you are specifying is what another person sees of you. You can have all the accuracy you want inside your system but there is no reason to communicate that accuracy to another guy because he can't see it anyway. Alright? So that's one thing I would like for everybody to remember.

Fidelity Tradeoffs: DIS requires many simulators. We are training or evaluating large-scale teams. That requires low-cost simulators which requires tradeoffs in fidelity. If we try to make all of these team trainers have the fidelity that a 30-million dollar simulator has, it's not going to work. You are not teaching the guy the initial skill acquisition, you are teaching him how to work with other team members. You don't need as much resolution or fidelity as you have in a 30-million dollar simulator. So keep that in mind.

The main goal of this workshop is to complete the draft standard for the PDUs. Whatever time is left in the subgroup meetings can be used to address other issues. That concludes my presentation. Are there any questions?

(Question) One of the things that I thought we had decided at the last conference was that this was going to establish a standard for heterogeneous simulators; that you would be able to link simulators of various fidelity. The implication of your last slide is that there will be homogeneous fidelity.

I guess I misstated that. This is a standard for interoperability. One simulator can be a multi-million dollar simulator and another can be a table-top computer. If we try to transmit the information that the larger system needs to represent all of its fidelity then I think we'd overwhelm the smaller system. We need to reduce the information to only that which the other simulators need to know. That is the point I wished to make. I am not saying that the high fidelity devices can't participate on the network. I am just stating that I don't want you to transmit the kind of data rates that normally would be transmitted to another full-blown simulator in the next room.

(Question could not be heard)

You are commenting that sensors picking up that resolution angle will require greater accuracy than can be seen normally. If that is true in the Emitter PDU then we may require more accuracy in the Emitter PDU than we do in the Appearance PDU. If a decision about that results from the subgroup meetings then we will implement it. The answer to your question is yes.

(Question) One of the things you mentioned that you were doing or planning to do in the future was to develop what you call an overall application or system description. That sounds to me like one of the things you should do early in order to define the system.

We have to admit, we were told to do a standard for PDUs and we did. In the process we found out that we had to define the application. Now I think we have enough information to define the overall application. We intend to go back and do it. Yes, I agree. It does seem a little backward.

(Question) You mentioned your work with the OSI protocols. (Unable to transcribe the entire question)

The question is: Are we going to enhance the OSI protocols to do multi-casting? Yes, that is our intent. We have listed the services that we need to have from OSI in order to achieve our goal. We are going to see if that is able to be done. We think it is, but not with an unmodified OSI. We think it is going to have to be modified. Hopefully the SPAWAR stack will be close enough to get us three-fourths of the way there.

(Question) You mentioned that these protocols would only be used for new simulators and that existing simulators wouldn't have to change. Only new simulators would have to use the standard.

The comment is that the standard would apply to new simulators and not necessarily require the old simulators to go back and change their protocols. If you've got a trainer that teaches mid-air refueling we don't expect you to go back and change the protocol but if you want to communicate between those two simulators and the outside, then you will need a gateway into the system that uses this protocol. Those two can continue talking to each other using their old protocol. Now if you decide to bring in a completely new simulator into that mid-air refueling device then you would need to go with the new protocol. Suppose you have a group of SIMNETs at Ft. Knox. One gateway could convert the protocol of the LAN to this new protocol for the longline to the rest of the system without modifying all those simulators. Only their connection to the longline would have to be modified. That

is what would allow interoperability with the rest of the system.

(Question) Speaking of things that work, I'm curious about how much flexibility there is in the OSI standard? It seems to me that in order to achieve the goal of the performance levels ... that you may have to compromise the OSI standards so far as it is still one of the standards. (Unable to transcribe the entire question)

The question was: Is there enough flexibility in the OSI standard to achieve our goal? That is the reason we are doing the study. We don't think you can take full-blown, unmodified OSI and go real-time. I don't think there is any question about that. There is going to have to be some modification. We do believe we would need a dual stack, one unmodified OSI for talking directly to a tactical computer, the other a stripped-down version of OSI for real-time data. OSI was not originally designed for real-time applications. The advantage of using OSI is the cost savings that will come as a result of the availability of inexpensive, commercially available software to support an OSI compliant architecture that is expected in the near future.

(Question) In the OSI model the lower core layers are basically communicating between one end and the other and the upper three layers are for the computers to be able to talk to each other. Do you envision that you will be developing, say, profiles for the upper three layers that sit on top of the lower core, to communicate?

Hold that question until the end of Chris Pinon's presentation.

(Unable to transcribe question)

The question is: If we are going to use OSI protocol, what other organizations are we working with that are involved with OSI? I attended the last two meetings of the Protocol Standards Steering Group meeting, which represents the people in control of GOSIP. Col. Shiflett briefed them five months ago about our work. I showed up three months later with a draft standard and they were shocked. Our standards group is coordinating with this Protocol Standards Steering Group. We have given them copies of the standard and they are reviewing it. So far we haven't received any negative feedback from them. There is a representative (member of the Protocol Standards Steering Group) here from Mitre who can answer your questions about OSI.

Thank you very much.

Overview of Standard, Christina Pinon

Today I'm going to give a brief summary of the standard, discuss recommendations that were made by the working group that met in July, and discuss recommendations made in the thirteen position papers that we've received since June 15th. We will begin with the summary of the draft standard. As Bruce mentioned, there are two documents:

- **The Rationale Document:** This document consists of the working group recommendations that were made before June 15th. We have included the working group recommendations, position papers, and also subgroup meetings that took place.

- **The Standards Document:** This document gives the requirements for the messages of the interaction of entities in a Distributed Interactive Simulation.

What is distributed interactive simulation? Bruce talked about what it is used for. I'm going to talk a little bit about what exactly it is. We did try to scope out DIS as a whole giving us something to work from. The definition that was developed is: an exercise in which simulated entities, generated by a number of interconnected simulation devices, are able to interact within a computer-generated environment.

In relation to the OSI reference model, which we've already talked about a little bit today, DIS as we have defined it in the standard, resides in the application layer of the communications architecture. There could be other application layer entities as well. This standard is independent of the underlying architecture. As Bruce indicated, we assume an OSI architecture beneath DIS, but we don't specify it as a requirement in the standard at this time until further testing is done. If you have questions concerning IST's work on architecture as well as other architecture questions, I would urge you to meet with the communications protocol group, in particular the communications/architecture sub-group where there

will be a discussion of IST's work. You can ask questions of the people who are actually working on it. They will also be discussing other architecture issues.

When we looked at distributed simulation we felt that there were four requirements for the kinds of information that need to be communicated between simulated entities. The first is entity information - information about each entity. An entity might be a vehicle, a unit of dismounted infantry, or anything in the simulation that needs to be distinguished. Also, information about the interactions between entities, environment information, and some kind of management function must be provided. Elements of entity information include entity type (you need to know what you are dealing with), the location and orientation of that entity, and the entity's appearance. We looked at two types of appearance: the visual appearance and the electromagnetic appearance. With entity interaction we are concerned with weapons fire, the update rate control (controlling the rate at which appearance PDUs are sent), logistics support such as repairs and resupply, collisions between entities, and electronic interactions (which would include the Emitter PDU).

Environment information: This aspect of DIS was not within the scope of the standard, but some of the information required by DIS is specified. This includes terrain, weather conditions that exist for this particular exercise, degrees of daylight or darkness, and other environmental effects which might include clouds, fog, etc.

Management functions, and Bruce mentioned these briefly, involve management of the network - control over the network devices, simulation management - where you have control over the simulation exercise, and performance measures - which deal specifically with data collection, fidelity requirements, and training requirements, etc. The current draft standard that was

sent out in June covers just two areas: Entity Information and Entity Interaction.

Let's take a brief look at the protocol data units. We are going to look at a just few of them to just give you a flavor for what's been produced in this standard. As we examined the PDUs there were a few things we considered. There is a PDU header and then the rest of the PDU. The header is actually part of the protocol data unit. The rest of it (the defined PDUs in the standard) is not a PDU in itself. We added some padding here and there to achieve 32-bit boundaries for efficient processing for 16/32-bit machines. Information that would help in filtering PDUs was put near the beginning so that, as the PDU is processed, it can be determined whether the simulator requires that information or not and, if not, the simulator can discard the rest of the information. The PDU header and these diagrams are just pictorial representations of what fields are included in the PDUs. These are also included in the standard, so if you have a copy of the standard you have these diagrams already. At the start of each PDU, the header specifies the exercise and the type of PDU that follows.

Related to entity information there was just one protocol data unit that we worked with and that is the Entity Appearance PDU. The Entity Appearance PDU is issued to communicate the type of entity that is sending the information to others and its location and orientation. Among other specific pieces of information, these are the most important.

We also included some higher order derivatives, acceleration and velocity, to accommodate high-order dead reckoning models. That's an issue that will be discussed later today. For entity interaction most of the PDUs we looked at are related to this area. PDUs are used to communicate the firing of weapons, detonation of the munitions, changing the rate at which appearance PDUs are

issued and logistics support (services, repair and re-supply) as well as describing collisions. Let's take a look at a few of those PDUs.

The Fire PDU is issued whenever a weapon is fired. It tells the people on the network what type of munition was fired, who fired the munition, where it was fired from, and information that they would need in order to represent that.

The Detonation PDU is issued whenever a munition has impacted or detonated. Part of what it describes is where the munition detonated and what exactly detonated so that each entity can assess it's own damage.

An Update Request PDU is issued to request more frequent issue of appearance information. Part of the information included in this PDU is who the change of rate is directed to and the criteria to determine the frequency of that update rate from threshold values.

The Service Request PDU allows an entity to request certain services such as resupply and repairs; others could be defined.

The Collision PDU is issued to communicate if a collision has occurred between entities. A reason to communicate this information is for the purposes of collecting information concerning the exercise. You can tell how damage was incurred.

Since the 15th of June, we've received 13 position papers and a number of informal comments. Those comments aren't included in the position papers (received at registration) but are very useful. We received a number from IBM and also from SYSCON. Because of these comments, the sub-group meeting in July was a very productive meeting. I'm going to briefly touch on what was discussed there. Included in my discussion are the issues brought up in the position

papers that have been resolved or at least addressed, and the recommendations that have been made.

In the position papers, several issues were presented. I won't list them all for you now. You are going to hear more about these this afternoon. Different position papers that are to be presented later made recommendations concerning these issues.

There was some question as to whether the June 15 standard handled certain issues in the right fashion. Position papers were submitted as a result. Some recommendations were made concerning these issues. These include:

- The use of quaternians, instead of Euler angles.
- Specification of dead reckoning algorithms - should that be done in the standard?
- The dead reckoning of articulated parts - should that function be provided for in the standard?
- Performing experimental evaluation and validation on the protocol - this is a very important issue and something that needs to be addressed.
- The standard specifies the use of fixed point numbers only. There was a position paper that recommended allowing floating point representation where it was natural and appropriate.
- There was some question about the use of bit-encoded attributes. There will be a presentation on that topic later. Recommendations for other PDUs were also made so that entities within the simulation can query each other for information.

The Subgroup meeting made some recommendations. Not all of them are included here but these are the main recommendations that were made:

- **Timestamps**: It was recommended that a time stamp be either absolute or relative; that one can choose either way and allow the least significant bit to indicate that. The timestamp should also specify the time at which the data is valid and not the time at which the data is issued from the simulator (which was specified in the standard). Simulators using the absolute time stamp should also support the use of a relative time stamp.

- **Binary Angle Measurement**, (BAMs): should be represented by unsigned integers. In the standard this was represented by signed.

- **Negative numbers**: should be specified as two's compliment.

It was also recommended by the sub-group that diagrams be used to clarify certain requirements. Instances of unsigned integers that are used for calculations should be specified as signed integers. Representation of tracers and amphibious vehicles is needed. These are some things that are on the agenda for the groups today:

- The subgroup recommended that the RADAR PDU be used in place of the Emitter PDU until a database can be developed. The Emitter PDU is based on a database of information which presently doesn't exist and needs to be developed.

- The use of guises, which is a SIMNET function, was recommended as an addition to the standard.

Here are some issues for further study (and these are issues that we hope will be resolved tomorrow):

- Articulated parts and how they should be represented
- How should electronic emissions be specified?
- The use of quaternians
- Production of a handbook to accompany the standard in order to explain in more detail what the standard prescribes.

That's all I have. Thank you very much.

Question and Answer Period

Bruce McDonald, Chris Pinon, Karen Danisas, Bob Glasgow)

(Bruce McDonald) This is the crew that put together the standard so we figured this was the logical group to be up here to answer any questions you have. And so, first question?

(Question) Could you expand a little bit on your goals or plans for validating the protocol? How you expect to do it and when you expect to do it?

(Bruce McDonald) As we mentioned, IST has some studies going on right now to look at OSI protocol. What we are going to test it with is the PDUs that we developed in the standard. We also have some companies that have asked to communicate back and forth with us, and so we have some simulators off-site that will be sending data in and out. In that process we will test the PDUs themselves, the communications protocol, and the fact that it is longlined with another device that was developed completely independently.

(Question) Would the validation include devices of differing levels of fidelity, perhaps what you might term to be high fidelity devices?

(Bruce McDonald) Yes, right now we are looking at a very high fidelity simulator and one that is currently scenario-driven that we are going to ask them to cut it in half and just use the user part of it. That is a medium fidelity trainer. We will be interacting with devices that we had nothing to do with creating and that's really what this standard is for. In theory, neither set of engineers who developed the trainers have to know the internal workings of the other device. They just have to have the protocol. We hope to find out if that is sufficient.

(Question) A long range question perhaps past that - What do you envision as the mechanism for validating that new offerors' products are compliant with this standard? Are you envisioning a test suite like you find for compilers or is IST going to be the designated person that if you can talk to IST you must be following the standard? What are your thoughts on that in the area a couple of years from now when this standard is being imposed on people contractually?

(Bruce McDonald) The intent is to copy what GOSIP is doing. The Protocol Standard Steering Group has developed a series of test suites. You take your protocol to NIST and run it through the test suite. If it runs, then it complies. We would envision someone developing test suites for this. IST is certainly willing to do it. We have not been tasked to do it. I believe that we may end up using a good bit of the GOSIP test suite as part of our test suite.

(Question) From time to time it has been said that the protocol document will not address architecture but only define protocol data units. I'm wondering what exactly is meant by architecture in that context? Does architecture, for example, mean where things are computed, and by what sorts of things? For example, what is the role of gateways and so on in supporting distributed simulation. Or does architecture mean what the underlying protocols are - what stack or suite of protocols is used to support the PDUs?

(Chris Pinon) I'd say that was part two of what you said. It's the underlying suite of protocols. I think we don't want to specify how each simulator manufacturer does their calculations in the computer's architecture, so I think the second thing that you said is what is meant by architecture.

(Question) Are considerations being made for intra-service and intra-trainer communications based upon different databases or different operating systems and levels of activity?

(Bruce McDonald) The group has a steering committee that has representatives from each of the services. We are using that steering committee as our avenue to the services. Since this standard hit the street we've gotten a lot more interaction than we had before. One of the major issues requiring cooperation with the services is terrain data bases and the emitter databases. That is one of the reasons this workshop is followed by the other workshop. It's because the other workshop gets into some of those database issues.

(Question) Are we looking at overlay type system on the basic system we are working on? Whatever trainer we've got by getting into the SIMNETing and whatever, is this going to be an overlay-type thing?

(Bruce McDonald) I'm afraid I don't understand the question.

(Question) Okay, I'll save it for later.

(Bruce McDonald) The intent is to specify a given terrain database format. One would probably get their terrain from Engineering Topographic Labs. Once everybody has the same set of terrain in their computers, they can interoperate. You are all going to see the same trees and the same buildings. We do have to standardize the format and location of the database. This would probably be provided by Engineering Topographic Labs.

(Question) Could you comment on how much of the PDU is driven by the assumption that each simulator needs to have or may need to have its own dead-reckoning models and whether that would be a requirement for the future for all simulators to play in the game?

(Bruce McDonald) I don't think that you could support the bandwidth without dead-reckoning. I mean, you can't tell everybody where you are every fifteenth of a second. So there has to be dead-reckoning and the intent is to actually put those equations into the standard eventually. You know that we all agree that if it's this type of vehicle, then you will use this equation so that everybody sees that vehicle moving the same way. And so, there are a few studies going on. I know that there is a project out at Williams Air Force Base where they are studying dead-reckoning in fixed-wing aircraft. Fort Rucker, I think, is using dead-reckoning on helicopters. So I don't see any way for this system to work without dead-reckoning.

(Chris Pinon) Yes, you can't have these without the dead-reckoning. The communications protocol sub-group is going to work on the issue of dead-reckoning because it needs to be further specified.

(Question) I was just wondering if you could address what sort of provisions are being made both within the protocol itself, and also in terms of administratively for expanding the protocol if need be in the future, and also to distinguish between different versions of the protocol if that issue arises.

(Bruce McDonald) We've always said that this is a protocol meant to grow. So like any other standard there will be a periodic review process, and when enough needs to be changed then we will issue another version of it. So, it's like any other standard. We don't see this as set in concrete, starting in January.

(Chris Pinon) Also, in answer to that, it would probably need to be specified in the protocol which version you are using. I think that was part of what you were asking and that hasn't specifically been dealt with in the standard, but I think you're right. If

there are going to be more versions of the standard then you need to specify which version you are using.

(Question) You stated just recently that there would be a standard database, especially for visual. This implies also then, perhaps, a standard image-generator. How are you going to deal with the politics, etc., of picking a standard image-generator for this out of the many that are available in this industry?

(Bruce McDonald) We are not going to standardize exactly how the thing looks in the Appearance PDU. We are going to say, "This is a tank with so and so capabilities and here's its turret". It's up to the image-generator to draw it. And I don't think it is a requirement that every trainer draw that tank exactly the same, or that fighter aircraft exactly the same, as long as it's close enough that the user sees it as the real thing. So I don't see that we're going to standardize exactly what it looks like. We are certainly not going to get deep enough that we'll select the image-generator. I don't see that in there at all.

(Chris Pinon) Part of it, I think, is his concern with correlation as far as line-of-sight and so on. And I think there is a correlation subgroup working on how you validate whether you've got it correlated right or not. I don't think that the standard is going to standardize a particular image-generator, and I'm not sure how they are going to resolve the correlation issues.

(Brian Goldiez) Let me try to address some of that, Steve. There are a couple of things we've come up with in the past couple of months. The underlying database source, whether it's 2851, or whether it's ITD, or whether it's whatever you want it to be, does not in any way assure that the rendered image correlates, because everybody polygonalizes differently. So, the approach we're beginning to look at the Institute is not standardizing on databases. It might have been a misnomer, but to try to develop

statistical measures of correlation so that if you need a level of correlation you can specify that. And that's the way we're marching. There is no way we're going to dictate to the General Electrics, or the BBNs, or the Evans & Sutherlands, how to build their hardware.

(Col. Shiflett) Let me put in my two-cents about that also. I'm Jim Shiflett. I'm a student at ICAF; I'm not a DARPA person now. But if you get to the point where you have to specify one manufacturer then you will not have a standard because it won't work. You have to deal with the subject of how you are going to incorporate a wide variety of CIGs into this endeavor. That was one of the tough questions - how do you have the same terrain so that there is a fair fight. There is a project that we've also been working on, another DARPA project. I think Chuck Benton is here from another contract we had where we had been looking at how can you have two different simulators with different CIGs operate on the same piece of terrain and have a fight. It gets back to the terrain database issue a lot. How do you specify that the underlying terrain is the same, and that the road and the trees are in the same place. I asked Chuck to bring down his IBM PC-based system which has a different CIG on it - much cheaper - and also to try to build a piece of the same terrain that we had in one of the SIMNET databases. Now, when you get a chance to go out there to the Institute and take a look at that, what I think you will see is that you'll have two different simulators hooked-up on the same network - interoperable - using the protocols that we have out in the field right now, working on the same piece of terrain, but the terrain looks a little different. But the hills are in the right place, the trees are in the right place, and the roads are in the right place, and now what you have to do is try to figure out if this is really going to be kind of a fair fight. I would encourage anybody else who is working on any type of machine like that, that you want to plug in to take advantage of what's down here at the Institute, because that is the reason it was set up - to assist

people in terms of dealing with those types of issues. The facilities that we have at Ft. Knox and Ft. Rucker are also available to try to address these issues. There is a very broad base so that you could hook-up five or ten different types of CIG systems and have a fair fight. There are still a lot of issues that are going to have to be addressed in terms of what is really fair, given that one of them is maybe a $20 000 simulator, one of them is maybe a couple of thousand dollars, and one of them might be a $2-3 million dollar piece of equipment.

(Question) I was wondering if that work that Chuck has been doing at IST is available for other people to use?

(Col. Shiflett) The answer is yes. We have been trying to push as much information out in terms of what's going on. I think some of the preliminary work that was done was in the last meeting where we talked about the SIMNET terrain database specification. That was an initial attempt to try to deal with the problem of how you define a piece of terrain. That will probably be a topic of discussion in the Terrain Sub-committee. But it's a difficult question, probably far more difficult than just the PDU question if you really want to have a broad-based standard that doesn't lock in one type of equipment. Also the Naval Post-Graduate School has done some work on the same thing as IST. George Lukes is both the Sub-committee Chairman and the person that I would recommend you talk to about the terrain database correlation issue.

(Lt.Col Sarner) I think there are basically three different goals here, two of which are legitimate for the standard. The three are: first, to allow simulators that are inherently different to operate together. The second goal, is to allow simulators that are inherently different but operating together to present a consistent picture of the battle. The third goal, which I don't think is part of the intent of the standard and shouldn't be, is to force simulators to be the same. I think you've got to preserve

the independence of the implementation because it is not, as I understand it, intended that the simulators will only be used in a network fashion, but we will take advantage of simulators that may be used in other applications as well and therefore will have competing requirements.

(Randy Sauders) I think that's a valid point and that's a point that I think has come up at all of these workshops. It's relatively a straightforward engineering problem to develop a standard to allow simulators of dissimilar requirements to interoperate and communicate with each other. Now, whether that provides valid training for the people who are in those simulators, whether that provides valid analysis of the engineering simulator for an experimental weapons system for use in deciding whether you want to go ahead and procure that weapons system or not, is not an engineering problem and is not a problem you can address with a standard. That's a problem that the user-community has to address as part of their decision where you say, "This is the exercise that I want to move out of the field and on to the planet SIMNET." And that's a decision not to be made lightly. The decision has to be made after some consideration and based upon some of the empirical results that, hopefully in a year or two should be available by people who have done that, because you can be easily misled by artifacts of the system since the current SIMNET tanks color-code the good guys from the bad guys. That is a feature and that was a necessary thing that had to be done to get those SIMNET systems fielded. But if you don't take that into account when you are deciding whether your TOW gunners are picking the right targets to shoot at or not, you're going to get the wrong assessment of your TOW gunners. You are not going to get a system that doesn't interoperate correctly, you are not going to get a network that crashes, you are just going to get the wrong answer. That is something we have to discuss, and it's good that you brought that up because that is a key thing we have to keep in all of our minds to make sure that we don't let the user community get misled about

the scope of the benefits they are going to get. They are going to get the ability to do experiments, but they are still going to need to think about those experiments and structure them correctly.

(Bruce McDonald) I agree with that, and one of the reasons we created the performance measures group is to address some of those very, very difficult to define problems. We are going to tackle a lot of those in the next day.

(Gary Depra) We have been working the UTSS Project for about a year and just to support this gentleman's comment, we spent most of our time working with the users and trying to define what the requirements are for training. In our opinion, you've got to start with the definition of aircraft missions and sensors aboard your own ship and work from there down to training tasks that you want to accomplish in a simulator. From there you go to simulation requirements down to, in our case, threat-models and threat data. And unless you take the user's definition of fighter requirements, in terms of what realism he needs, I think you are in trouble. We are very, very concerned with mistraining. If I've got the solution that allows a pilot to, for example, "break lock" of a tracking missile that works because of the simulator artifact, I think that is completely unacceptable. I am concerned particularly with this problem. I believe dead reckoning is going to give us those types of artifacts.

(Bruce McDonald) I don't know about that but we definitely need tactical people in performance measures groups. We want to know what you would do if you saw that. If you wouldn't do anything, then it's probably not needed. If you would key on that, then it's got to be in there and we will specify that. So I definitely need help from tactical people in the performance measures group.

(Question) I wasn't at the previous working groups and so my organization, The Air-to-Air Combat Simulator at Luke in Phoenix,

does not have, to my knowledge, a draft of where the standards are at this point. So I don't know their contents exactly, but I have one question, after sitting here and looking at the presentation on the data packets. They seem to be numerous, for one thing and rather lengthy in their number of bits. So I am curious, even though you stated architecture may not be the concern at this moment, as to what do see as the underlying structure, given today's technologies, and given that something like ETHERNET that is 15 years old may be a defacto standard. But what are we looking at for throughput speeds and things to handle that amount of data?

(Chris Pinon) Go to the Communications/Architecture Sub-group. They will be talking about that specifically. We really did not deal with that. We said we are going to stay in the application layer and talk about what kinds of information needs to be communicated, realizing that there are some assumptions that need to be made concerning the architecture. That was pointed out in some of the position papers but the Communications/Architecture Sub-group is handling that issue. They've already gone into some detailed discussions concerning that. Specifically, the March meetings laid out a process for developing the kind of architecture that can handle those messages.

(Question) It would appear though, even if you talk in the applications/software level, that you've created a rather man-hour intensive program of having to take the communications as they exist in your own unique simulators of whatever fidelity, that is, and do a lot of re-manipulation just to get the data in a form the network can use. So, I mean it's not just an architecture question. It looks like there is going to be a rather extensive bit of applications also involved in making the data come out in the shape and the size packets desired.

(Brian Goldiez) Let me address a couple of those things via some other work going on at IST. First, the packet size that you see up

there is not fundamentally that much larger than the existing SIMNET packet sizes. But what I am saying is that is the only thing we have to go on. That's the best source of data information that we have been able to get in the last year. And it works. The other thing is that in terms of underlying bandwidth, there is some work going on at IST. There is some work going on I am sure at other places that looks at not only ETHERNET, but some token ring implementations. We are going to be doing FDDI with XTP that are an order, or two orders of magnitude higher, in terms of bandwidth, than the existing ETHERNET. We are in the process of implementing tokens right now.

(Bruce McDonald) The PDUs are probably bigger than they could be because we were trying to stay on 32-bit boundaries. We were told that the experience had been that if you violated that boundary, it took more time to process the PDUs than it does to take them in. And so you really have more trouble talking to the network than you do overloading the network. We very consciously created these PDUs so that you can read them very quickly and process them very quickly. That adds a little padding and makes it a little larger, but processing the PDUs appears to be more of a problem than overloading the network.

(Question) Do you eventually expect the standard to address voice communications?

(Bruce McDonald) Right now we are not looking at that. We do have studies going at IST to look at encoding voice and putting it over the network, but at this point in time I don't think it's envisioned to be part of the standard. If the Steering Committee decides we should, then we could add it, but right now, no.

(Brian Goldiez) I would just like to comment on that for a second. There are some application-layer developments going on in various places that speak to, for example, the voice communications issue.

For instance, Larry Goldberg is here from CECOM. They are working on something for SINCGARS in that respect. Basically, Larry will have something to share with the group on communications architecture to bring voice into play as a, perhaps, "assister" application layer entity.

(Jim Hammond) A year ago the Terrain Database group was not much interested in oceanographic data and you mentioned expanding the terrain database. We are down at Naval Oceanographic Office and Chief of Naval Oceanographic Command and are interested in the standardization of the oceanographic database. We were wondering if any of this was going to be something that the Navy was going to come up with by itself, or if DARPA and the TRI-Service group would be interested in looking into an expansion of the terrain database to cover the oceanographic area?

(Bruce McDonald) That gets into a philosophical question, whether the water is terrain, or atmosphere, or environment. To me it's environment. We are anticipating that the terrain-based group in the environment will address that. We haven't had enough Navy people involved in the past to address that. We were able to pull in some ASW functions and things like that, but we didn't have enough expertise in the past to get into the environment underwater. We are very glad to have people here to help us in that area.

(Question) As a follow-up to Al Kerecman's comment, with the help of BBN we have already developed a radio simulator which resides on SIMNET, and simulates SINCGARS radio communications in the same type of protocol structure as the vehicles themselves and exist in the same environment. The coding that is on the protocols tells you that it is a radio simulation and it includes a simulation of digitized voice that is on the SINCGARS radio. That would be very easy to port over into the work that you are doing. It would give

you the capability of having radio networks on top of the other simulations that you are doing.

(Bruce McDonald) Bring that to the sub-group meeting and we would certainly like to look at it. I am a little concerned about the communications filling up the bandwidth that we have for the rest of the stuff. So to me that should be on a parallel line. Put them on a separate line and then it could be accommodated without any problem.

(Question) To shift modes a little, going back into the requirements definition process, we are talking a lot about distributed architectures, or network systems as a requirement. I'm sure DARPA anticipates a groundswell of those sorts of requirements coming up. That is one of the reasons for this investment on all our parts. How would you differentiate or compare the concept of simulator networking with that of combat mission rehearsal? I would just like your thoughts on that as a group.

(Bruce McDonald) In mission rehearsal, if you are going to teach an individual how to work as a person by himself or maybe with just a small team, you can teach without network g. But if you want to teach or practice a mission that is going to combine several Services with multiple vehicles under different command structures, I think you are going to have to get into a DIS-type environment. I think we found that out in Grenada; we found that out in Iran. So I think you probably could have some initial operator training in dedicated simulators but then you are going to have to go into a DIS-type environment for the final training for the overall teamwork.

(Question) Very few missions are done by one platform. I think you can probably go through a couple hundred years of our history and find very few instances where only one platform, was involved in a major battle or war. At the very least, even if it was just

one platform it had a command & control input that was external to that platform, which is also part of the engagement. The idea of combat mission rehearsal implies to me that you need to have networking of those disparate functions that compose the mission: more than one platform, more than one platform type, more than just armaments and weapons systems, but also the command & control structure necessary to support it. It requires training just like the crews do. I see combat mission rehearsal as being networking.

(Bruce McDonald) I think the majority of training will end up in networking. I really do. I think there are some trainers right now that will not teach the teamwork part of it. They can be used for initial operator training or skills training, but once you get into the real nitty-gritty of it you are going to have to go to DIS-type environment. I think most of the training will occur there.

(Question) So will combat mission rehearsal?

(Bruce McDonald) Yes.

(Question) I would just like to add one comment to what I said before. The purpose of the radio simulation is to provide the command & control functions, so that fits in with the comments just made.

(Bruce McDonald) I can see if you have a small number of entities, you could probably incorporate your communication in there and cut down on your number of lines required. If the number of entities is going to load up your network, then you probably would have to have a separate channel. I know that for one of the discussions on how you handle priority, one of the proposed solutions is that the command & control communications have a separate channel so that it doesn't get stepped-on by a bunch of appearance PDUs. So that

would get it through without any delays. That is being addressed by the Protocol Groups.

(Question) You mentioned a little while ago that you envisioned in the future that there would be an evolutionary development of the standards, and that as the need was perceived, new versions of it would be released. Has the organization here taken the basic position that new releases of the standards are going to be backward-compatible with the old one so that there is not an enormous redevelopment problem every time you release a new one?

(Bruce McDonald) One of the criteria for a modification is that it doesn't kill everything you've just done. It's probably not going to be worth it if you go to square one every time. I can envision an update to the standard that, if you comply you can do some more sophisticated training and that if you have the old standard you would not be able to sense that environment. So I certainly picture each modification to be compatible with the old ones but they won't be able to play in some of the new capabilities.

Distributed War Fighting Simulation, Major Jim Wargo

Good afternoon. My name is Major Jim Wargo. I'm from the DARPA Tactical Technology Office and I'll be presenting the DARPA program in **Distributed War Fighting Simulation.** The actual program manager is LtCol Mark Pullin, who hasn't yet arrived. In light of that, I will try to cast my remarks in terms of the remarks that were made by the Honorable Chris Jehn this morning. Many of my comments will echo his.

I am with the Advanced Distributed Simulation Technology Program, (ADST). You are probably familiar with the acronym, ABS -- (Advanced Battle Simulation). You might have heard the acronym, DSNT -- (Distributed Simulation - Networking Technology) and there are probably a half-dozen acronyms that you are also familiar with. All of them convey one particular image: the interoperability of all levels of simulators and simulations. I think the words that Mr. Jehn used this morning were most appropriate. He used the phase **the full realm of artificial environments.** But I guess the best single adjective **Seamless Simulation.** Seamless simulation is the idea that we should be able to link together real equipment simulators, whether actual equipment or computer-generated entities and simulations -- war games that are in existence. From individual, through the higher levels of collective tasks, up to the highest levels of joint-command and control, regardless of the branch of service or whether it's U.S. or allies, **Seamless Simulation** is the stated DoD goal that Mr. Jehn came up with this morning.

This is the same information presented in a slightly different format. You might have heard the terms horizontal expansion and vertical expansion. If there is any appropriate group to talk about horizontal expansion of simulators, it's got to be this one. Many of you either perform some work or accomplish some work in that matrix which was presented on the previous slide. And that's

a definite advantage. There are a lot of people doing work in Seamless Simulation. On the down side, I don't believe that anyone has a complete fence around that, and no one can really say exactly what work is being done. The bottom level concerns simulators: SIMNET-like simulators, high-fidelity simulators, lower-fidelity simulators, and their interoperability. The development and issuance of a draft standard, as you propose to do, is a necessary first step. We can't underestimate the importance of it. The development of a Draft Standard will go a long way toward establishing the interoperability of simulators. There is additional work that needs to be done. Some of the questions that were raised earlier are: How do you level the playing field between a low-fidelity simulator and one that is necessarily a high-fidelity simulator? How do you draw reasonable and fair conclusions when those two types interact? How do you account for the various technologies that are involved?

On the next level we have existing war games. Someone asked this morning, when the war-gamers are going to be involved. Part of the Distributed War Fighting Simulation Program involves the interoperability of corp-level simulations. But DARPA is not alone in that work. PM TRADE, in particular Mr. Pat Spangler, is involved in detail linking JESS with JESS, and JESS with other corp-battle simulations. Like I said, there is a lot of work being done and if I haven't mentioned yours it's probably because I'm not aware of it just yet.

On the highest level we have the Command & Control networks of the NATO allies: primary subordinate commands, the major subordinate commands, and the Supreme Headquarters Allied Powers, Europe. DARPA's stated goal is to, within the next five years, introduce into the DoD inventory a single-simulation that connects all echelons, all allies. Comments that Mr. Jehn made this morning make that a DoD goal.

The specific purposes of the Distributed War Fighting Program are shown here. The first goal is to link the primary and major subordinate command headquarters of the NATO allies. A complimentary goal is to link those headquarters with the war games developed at the Warrior Preparation Center. As you can see, those goals are complimentary to a seamless simulation. If we can somehow orchestrate all the efforts together, I believe we will accomplish these goals in a more timely fashion.

The bottom bullet is of particular note in that if we do achieve the goals of Distributed War Fighting Simulation, that is, linking all the Allied Command Headquarters together, then how do we draw the distinction between simulation and real world? We have converged reality with simulation if in fact the actual Command & Control networks that we are going to use in simulation are the same ones that we are going to use in the actual war fighting. So from the top down, we enforce realism into the full realm of Distributed Simulation.

Like SIMNET and SIMNET-like technology, the advanced distributive war gaming system program also takes advantage of advances in the state-of-the-art. The program takes particular advantage of multi-media conferencing, high definition displays, more reasonable behavioral representation of threats, opposing forces and friendly forces, and finally, improvements in processing capability. As we take advantage, the state-of-the-art will move from the demonstrated capabilities in Allied Command-Europe Exercise '89 to the proposed exercise in ACE '92. It's still a few years off but we are moving on schedule.

This slide is in one respect a repeat of the two previous slides. But this slide serves to emphasize the fact that the Distributed War Gaming Simulation system in Europe takes a slice of existing SIMNET and SIMNET-like technologies which are on the ground, Allied capabilities which are on the ground, and future

enhancements planned by DARPA, and also those existing games at the Warrior Preparation Center. What is probably obvious to you now is that some of the remarks that Mr. Jehn made earlier, (**where do we go from here?**), the extension of the protocols and the standards to higher levels. These solid lines represent what's required in the manner of standards to piece the entire system together because right now they can't all communicate with one another, at least not to the degree that we would like.

Now this is probably sinful talking to the folks who will be establishing the standards, but there are obvious advantages to extending the standards to Distributed War Game Simulation. Many of them are shown here. It should be emphasized though that it adds flexibility if we can utilize existing models, refine them, improve upon them, and perhaps eventually replace them. So the DWS is not really a threat. What we are attempting to do is to accommodate existing models for a mutual benefit.

This slide simply represents an example of how such a network would use those protocols. The slide shows several different ground models, GRWSIM, some generic Allied ground model, communicating over some generic token ring, communicating all the information required for the other games to recognize that there was an indirect fire attack. That information includes locations, local time, game time, lethality, and Lankestrian units to determine the outcomes of the battle. The effect of the protocols and standards will be to allow each of the opposing games to modify its database to account for this attack. The idea is that it is not necessarily just a combat model but we could have logistics models, air models - you name it.

The last two slides represent the Distributed War Fighting Network. This network is not yet in existence in Europe but the plan is to develop the land lines with a redundant satellite

capability to allow for redundancy in remote locations. The plan is to have this installed for ACE '92.

These three solid lines represent the backbone that is actually installed in the Defense Research Internet that Mr. Jehn referred to this morning. The dotted line represents the extensions that we should have completed by November. So, I guess if there is a bottom line it is that there is now a stated goal at the DoD level for Seamless Simulation for all levels. There is an infrastructure in terms of the network that is on the ground now and the network that will be developed. There are existing networks of simulators. There are existing simulations. All of the tools that are necessary to develop Seamless Simulation are there. All that is necessary now is to extend this standard beyond the point where we are now. Thank you.

Technology Push Requirements Pull: A Navy Position, Tom Tiernan & Kevin Boner, NOSC

This presentation is presented in two parts. I'm going to speak during the first part. The first part provides information on an approach to achieving a Navy position with respect to the standard. The second part, which will be presented by Kevin, discusses some of the specific elements in the fields in protocol data units themselves.

In SIMNET our initial approach was to think of SIMNET as a cross-pollination of technology and requirements. From the Navy perspective, the hard part for many people, that we talked to was focusing on SIMNET the product, versus SIMNET the technology, and SIMNET the requirements. In terms of Economics 101 the way we approached it was: This is why they have answers looking for problems, and on this side are questions seeking solutions. And that's how we got involved with the proof of principle that I'll talk about here in a few minutes. So we look at the technology push coming from the extension of what was originally done in the product SIMNET and the standards for the Interoperability of Defense Simulations, and we look at the requirements from the Navy perspective. I think enough evidence exists in the Navy to indicate that there is a requirement for networking these and other systems together. There is a Battle Force In-Port Training operational requirement. They currently conduct BFIT exercises up and down the East and West coast. However, it is documented that the methodology of software transfer is inadequate to handle what they want to accomplish. So we looked at who has some answers versus starting over and the technology push came to mind. There has also been some work done by NAVAIR with respect to Road Map, and highest and best use of funding to upgrade the ranges. There have been some concepts by the Pacific Missile Test Center and by the Naval Weapons Center in China Lake regarding World Range. There's an OPNAV instruction 1500 Series that talks about a surface

warfare strategy where we need to look at other means besides major schoolhouse trainers to train people to do the jobs that they're outlined to do. Certainly, conference papers that are presented in symposiums and conferences such as this indicate that the trend is there to provide networking. There are very broad applications to what we call the warfighting network, but I think as a whole the requirements exist, and the technology is being pushed in that direction, which is good.

Concerning the process that was described previously where the standard is progressing down this path: we're probably at step one and step three in this part and this is going to come later in the calendar year and in the first part of next year. But when we get to this part, to resolve comments by the Department of Defense, the issue becomes as the individual walks across that bridge to seek approval, he looks at the Puzzle Palace and wonders who is the individual who going to get that. There is a significant difference between walking across the bridge and giving it to someone saying, "Yeah, I got it. Put it over there in that pile that goes in priority nine and I might get to it." The other aspect is working the way that we are working, which is primarily this group, feeding information up here when this individual comes walking across the bridge, he gives it to the right person, the policies have all been set, "thank you very much, I will execute", and the information will then flow down to the people who can execute. There are a lot of people who like the concept of networking and are doing a lot of work in this area. They have the equipment and the requirements perceived, but they need instructions and guidance from OPNAV. I think maybe the Dod process also comes down here to NTSC to integrate and standardize these protocols for new simulators, but from the Navy's perspective we also have a large number of existing training systems that we'd also like to bring into the network which may fall outside the realm of NTSC. And those guys such as NAVAIR, SPAWAR, NAVSEA need

written guidance and policy statements from their OPNAV organization.

The people who got together from the codes listed on this view graph are a very small sampling of the people who may be impacted by this, but nonetheless, we wanted to find out what the pulse of the thinking was with respect to the standard, and how they might be impacted. We got pretty good feedback. We need to continue this process. I think the standardization process that we are going through is going to force the Navy's hand, which is good. I think the consensus of the group was that there is a requirements pull. There is some documentation, such as the Battle Force Tactical Trainer operational requirement and Tactical Combat Training System. Those two things need to talk together as well as to other training systems that exist and are being developed. And so the process for achieving a Navy position is something that we are pursuing. I'm not here today to tell you what that Navy position is, just to relay the fact that we are pursuing that effort.

On this view graph there is a little bit of background information on a Battle Force In-Port Training SIMNET proof of principle which was conducted in April of 1990 this year. We accomplished some of the things that are outlined in the view graph. The point here is that it was not an end item, nor did it continue to be, and we need to continue to address the issues because some additional issues were brought up. For the work that was done in the BFIT SIMNET proof of principle, the NAVAL Ocean System Center was the systems engineer, Kevin was the lead systems engineer, and I was the program manager, Mike Newton wrote the software. Mike Healy did most of the programming from a little company out in San Diego - ETA Technologies. We were able to work some of the existing simulators in Ft. Knox which were driven by Marine Corp pilots. Some of the air simulators in Ft. Rucker were driven by Marine pilots. We had the USS WASP, which is a Navy

ship, pierside in Norfolk. For operational equipment, we had 1965 MIL spec Consoles at Fleet Combat Training Center, a MK86 Gunfire Control System, and a Tactical Flag Command Center.

It was mentioned earlier that LCDR McBride is going to give a brief at 1700 with respect to BFIT. That's partially correct, and partially incorrect. He's not going to give a brief. We have a 10 minute video, that illustrates what we did at the BFIT SIMNET Proof of Principle, that will be shown, at 5:00 p.m. if you are interested. Bill Harris has asked us to address security. We're going to give a separate presentation from this in the Security Working Group. That presentation will talk about the lessons learned and some of the issues that came up in the Proof of Principle. We also did some work with the Protocol Data Units. Since that time Mike Newton, Mike Healy, and Kevin Boner have put together some issues that Kevin is going to present now, which were discussed at the Navy meeting on the 17th and 18th of July, in Washington.

(Kevin Boner) What I would like to do is briefly bring to you some of the concerns that we have from our meeting in July. Basically we are going to look at three of the protocol data units today. The first one is the Entity Protocol Data Unit. Going through the standard, the Entity Protocol Data Unit primarily in the entity type defines each of the ships by ship class. One of the things that was brought out was that we would like to have it via a particular ship because different ships in a particular class have different characteristics. Another thing that was brought out is the Articulated Parts. With the Articulated Parts as it is defined right now only one articulated part is defined. What we'd like to do is have Sub-articulated parts if possible. Primarily from the standpoint of a gun turret you have multiple-moving capabilities. Under the Entity Appearance, the surface domain for surface platform, some of the things that came out were that right

now you have a bits characteristics for "on" and "off" for flaming. One of the things that came out is that we would like to have a location capability as well. We would also like to allow for multi-lighting systems aboard the carrier systems. We would also like to propose to eliminate the muzzle flash bits since they are generated by the Fire PDUs.

With the Emitter Protocol Data Unit - some of the things that came out there, there is no direction capability for the emitters, for instance a fire control radar. What we would like to do is allow for direction for the emitter. We also looked at the database information, and we would like to allow for more pointers into the database from the characteristics on the EW, primarily, and also the databases. It was mentioned earlier this morning, that one of the things that came out from our meeting is that from a Navy perspective there are several databases out there. Probably each system has its own database and so we need to also look into that area as well.

On the Resupply Protocol Data Unit you cannot specify the type and amount of fuel and ammunition that you want, so this is a recommendation on our part and also on the resupply. If the resupply is canceled there is nothing transferred, and we would like a partial resupply.

Earlier this morning, Jim Hammond brought up some of the other concerns from our meeting. We would also like to, instead of looking at the terrain, look at some of the ocean characteristics. The environmental issues like type of weather, sea state, and water temperature which are, from a Navy perspective, important.

The other thing that we learned from our Proof of Principle is from a Navy perspective - security is an important issue. Navy tactics are classified. When we tie to SIMNET the ARMY tactics are unclassified, and this is an area that needs to be addressed. One

of the other concerns that was brought out is from a large scale exercise we would like to propose an Aggregate PDU. What we're talking about here is from a Navy perspective. One of the things we are looking at is to go up to a 2000-track capability. With SAFOR, or things of that nature, we would like to have an Aggregate PDU.

Are there any questions?

(Question) When you are talking about the different systems that you've been looking at from a Navy perspective, I was wondering if you had taken a look at ENWGS Enhanced Naval Warfare Gaming System that is distributed between Hawaii, CONUS, and England, and if that had been part of the things that the Navy had been looking at?

In the Proof of Principle we did not include the Enhanced Naval Warfare Gaming System. We are familiar with it. What we tried to do was bite off something that we thought we could accomplish in a period of performance that was not two or three years downstream. So we took off a bite-sized chunk. At that point one of the biggest proponents of trying to get fleet participation was from the Atlantic Fleet at the Fleet Combat Training Center, so we included them. Inclusion of the Enhanced Naval Warfare Gaming System, I think, fits in line with the way that the standard is moving. If this standard were to be put into effect and you wanted to have a battle station "B", something like the Enhanced Naval Warfare Gaming System, you could develop the network plug to do that. We didn't specifically exclude them, we just did not include them in the work that we've done to date.

(Question) You were talking earlier about systems other than schoolhouse systems that needed to be looked at. Could you give me an example of what you mean by it?

In the Proof of Principle we used the Combat Simulation Test Set made by Litton Data Systems for LHD-1. It's an embedded training capability. It has it's own RADAR simulators and we used that as an embedded training system. Another one on the Fleet Combat Training Center Atlantic has a video signal simulator which actually puts the RADAR displays on the scope. We had to stimulate that via the PDUs so that the crew could react to that with their operational equipment and send their tactical data such as LINK-11. Future thoughts include the SQQ-89 On-board Trainer and ASW (which also includes some EW stimulation as well). So what we are really trying to see is whether the platform that we work from exists in development and planned training systems.

(Question) Could you expand on the aggregate PDUs that you just mentioned in your last item?

Basically, what we are looking at from the aggregate PDU with the 2000-vehicle capability is, the requirement being that, (if you are sending the vehicle appearance PDUs out over the net). the SAFOR capability is that you've got groups of ships acting with the characteristics, so you would group those together. So that is what we are looking at.

(Question) It sounded like some of your data was time-critical that went over the network. Did you have to do any kind of time-tagging or synchronization of the data?

We did not utilize time-tag during our BFIT Proof of Principle.

(Question) Okay, you didn't have any time-critical elements or time critical...?

Well basically the problem we ran into from a tactical situation, from a Navy standpoint, was that we could not actually

utilize some of the Navy tactics. So, it ended up being more of a engineering-type Proof of Principle, so to speak. So we really didn't get into the time-criticality.

Top-Level Standard Implementation from an AIR FORCE Perspective, Mr. Tom Hoog, ASD/ENET

Some of the things I'm going to say are going to sound a little bit negative. They are kind of negative, but stand by until I get to the end. I talked to a number of participants in past conferences, and discussions this morning and what was presented this morning, make some of the things that I've addressed begin to become a little more clear.

Well, basically today the current state is that we have a draft standard. A proposed network is described, however, the extent of application does not appear to be clear. There are some issues that have been identified and some of these have been closed. Specific PDUs have been incorporated. This all represents some good work. It shows a lot of people have put some time and effort into solving specific problems. A number of opinions have been expressed, some of them varying, but there has been some convergence (as was indicated) earlier on some of the issues. There are, however, some new issues that have been identified, and some previously identified, which do remain open. One thing that I heard this morning is that it appears that some things are not within the current scope of the effort that has been defined. I'm not sure what that is so it adds a little more confusion in my mind. Some of the things dealing with environment, weather, terrain, and somewhat the electromagnetic environment are some of the things that still need closure. There have been various degrees of participation by the Services and I think some of this is evident in what has evolved in the current draft standard.

I do have a concern. Something seems to be missing. We've got lots of details, a lot of people working the nuts and bolts but there is still something missing. We've got a broad, overall statement of what it is we are trying to do. But when I read the standard there is still, to me, something missing. It seems to me

we have a solution but we don't have a fully defined problem. Now, let me explain what I mean by that. As I mentioned, most of the work seems to be dealing with the nuts and bolts. Everybody has a little piece of the picture and they are putting it on the canvas. But what is the end picture supposed to look like? Are we all working to fill in the same picture? And I think some of the discussion and some of the questions that I have heard indicate that maybe we are not. We all have a vision and it is fine to have visions but are we all working to exactly the same goal? If this standard is to truly be used Dod-wide, all the Services must participate together in essentially using their own needs plus inter-relating together to establish the interoperability needs. And there must be some rationale about how they are going to do this or not do certain things. That will help us, I think, in resolving or coming to closure on some of the open issues.

Some questions: What are the training requirements? As I mentioned, we have a broad overview of very top-level requirements and we've got some very diverse needs among the Services. But what are the training tasks? Exactly what do the users have to do, the people who are being trained? To what extend must the Services simulators interact? You can have multiple training without necessarily having a lot of interaction. This has also been addressed today. Are there alternative solutions to just interconnecting simulators? There might be some cheaper ways rather than maintaining long lines and other means of communications. What are the performance requirements? Once we get the training requirements then we can derive some performance requirements. What has to be done? Who has to do it? Is there going to be a single standard or multiple standards? It appeared to me that there was going to be a single standard and it was going to be an attempt to satisfy everybody's needs. Maybe that's not quite possible. Maybe that's not what we want to do. Maybe there ought to be multiple standards. We need to address who is going to play, how much they are going to play. And as I mentioned in the

beginning all this has to lead back to what the real training requirements are. What are the tasks to be performed? We need to get closure on those issues that are still open. We need to resolve them. I think we need to have closure plans so that we can see where we are headed and what the time table is for bringing all the pieces of the picture together. Is there a method for taking care of new issues? How are we going to resolve them? As we get into it, we are going to uncover new things, things we haven't thought of before. New requirements are downstream. Is there a process in place for handling these things? And just how are we going to implement the standard? Who is going to take exactly what action to reach closure? Who is going to maintain the standard? We heard some of what the overall plan is this morning. What systems are going to use this standard initially? Are we going to start with something simple and go to something bigger? How are we going to mature the standard; how are we going to test the standard? We need to validate what it is that is going to be done. We don't want to go out and just use it for a first-time application on a set of simulators.

A few recommendations: I called it the Systems Engineering Process. That's an overused term, but basically I think that's what we've got to do. We've got to go back and fill in the pieces. We've got some very top-level requirements, and we've got a lot of things down at the nuts and bolts level. We need to fill in between the top level and the nuts and bolts level. We need to find out what the specific training requirements are. Is everyone participating in an interactive scenario being trained, or are some of them just supporting the training of others? We need to derive the top-level performance requirements. I think there can be a lot of controversy if we don't go through this in a logical manner. All the Services need to do this; then we need to join together to find out what is common, just how far apart we are, and what we see that needs to be done. How far apart are we on what the real training requirements are, and thus, what are the performance

requirements needed to satisfy that? From that we just get into the normal implementation process and then develop a detailed design. Another thing we ought to address right up front is just exactly what are the validation requirements. How are we going to test this? Everybody needs to know that up front. You have to know how much you are getting into this before you start. What is the investment needed over the whole life cycle in order to participate? We need to document what our plan is and make it visible for everyone. Again, we've got some indication of what the top-level plan is. We saw some of that this morning. But we need to press further. We need to get down to exactly who is doing what, and what is their timetable for getting each piece done to meet the overall milestones.

Maturation of the standard: We shouldn't go out and try to do the whole thing at once. We need to start with something simple and then build on that. There has been a base, as indicated, starting with SIMNET. That's fine. But we need to take this product as a new product and use it as it's stated, then build on that. As I indicated, we need to look ahead to how we are going to carry out this thing through its life-cycle now before we get too far downstream have to back up and ask, what we really trying to do? Specific actions which I see needed right now are to define and publish the road map, in detail, and give everybody something to look at. It will make it visible so that we can all see and all work together so that we get to the same final picture. It seems like a group might be needed to do some of this, an integration group. There are a lot of issues that may be common that have to be addressed at the groups and sub-groups that perhaps ought to be tied together in another group. It might be possible that the steering group is performing this function. It's not clear to me. But there needs to be a way to address the higher level, top-level issues that just don't quite seem to fit any one of the groups or sub-groups. There needs to be a place for them to enter the system and be addressed.

Some of you may think I've stood here and kind of trashed the proposed standard. That is not my intent. I certainly was not thinking that way. As I said, there has been a lot of good work done. A lot of things, have been resolved. But we need to fill in what it is we are driving towards. I'm just trying to make sure that we get a clear look at where we are headed and when we are going to get there, throughout the whole community. And I think now is the time to make sure that we are on solid ground. We don't want to wait a year or two from now and then have to back up. We still have time to fill in some of these missing pieces. We need a clear statement of what the training requirements are, followed by a good understanding of what the performance requirements are, and then we need to carry out those performance requirements. We need a complete program development plan that will show us how to fully develop and implement this operability standard.

The Advantage of Using Quaternians Instead of Euler Angles for Representing Orientation, Dr. Jerry Burchfiel, BBN

I'm Jerry Burchfiel from BBN and I'm going to try to convince you today that we should be using quaternians instead of Euler angles. During the course of my talk I hope to convince you of three major advantages that quaternians have over the Euler angles that have been recommended in the Draft Interactive Simulations Standard. One is that we can eliminate numerical integration and all the error that comes with it. We can eliminate singularities that I am going to show you on some of the equations that are part of Euler angles and greatly reduce the computational effort. So my recommendation is going to be that the DIS standard should use quaternians instead of Euler angles to represent orientations.

Let's back up to the basics first. What is it we are trying to do here, dead-reckoning or remote vehicle approximation? The basic idea is that the receiving vehicle extrapolates in a simple way, the movement of the other vehicles on the battlefield between hearing new appearance update packets from them. It continues stepping them ahead on its visual displays, that is, it will continue showing the other vehicles moving smoothly along in-between hearing updates from them. The big benefit that gives us is that it greatly reduces the network traffic. We don't have to keep repeating ourselves fifteen times a second but we only have to report when there is some visible change in the behavior of a vehicle. Even more important is that the packet processing load on that receiving simulator is greatly reduced. The real bottleneck that we experienced in SIMNET was how many packets that a receiving simulator could pull off that network and process in any intelligent way. In SIMNET today we are doing forward extrapolation of the last position using a constant velocity. This is called dead-reckoning. So, if I have a vehicle that is moving down the road, we continue moving him down the road at a constant velocity, or an airplane moving through the air moves in a straight

line at a constant speed. Orientation is constant so we don't show any rotations. We have a new opportunity here, now that we are opening up the issue of how to really do remote vehicle approximations, and that is that we could use a higher order model for aircraft. There, the aircraft maneuvers are often well-modelled by constant rate turns. In particular, the way we could describe that is that we have a turning rate omega around an axis "N". I don't know if that shows up very well (on the view-graph) but the intention is that this is a unit vector and the velocity is constant in platform coordinates. I have brought a real-time, full color, 3-D simulation of a moving object with me so I can show you what I have in mind. Moving the model around like so, you can see that in all of this is a constant rotation rate around some axis. Maybe doing this kind of a thing or whatever with a single scalar-rate omega and a single unit axis "N" we can reasonably model that class of maneuver. One final point is that what we are looking for here is in order to get realistic appearances in these receiving computer-image generators. We want smooth motions. We don't want it jerking this way and that. We don't want to see any angular acceleration between these updates. We want to see a constant rotation rate in motion.

Now what's in the spec today is Euler angles. What are they? They were defined by Leonard Euler in 1752 to specify the orientation of a solid body. They are always three angles and there are twenty-four different ways of picking those three angles. You can rotate around any three axes as long as you don't make two successive ones the same axis, then you've got your three degrees of freedom. In the choice made in the DIS standard which is the one that is common in British Aerospace tradition that has then been handed down to U.S. Aerospace, is called Tate-Bryan angles and they are also called yaw, pitch, and roll. The idea there is if we start with an object in some fixed orientation with respect to world coordinates, then I can transform it by doing a yaw, a pitch, and a roll. The combination of those three motions gets me to the

final orientation of my vehicle or platform. This was, as I said, the recommendation that came out in that July Draft Standard.

Now the Euler angles have a number of problems. One of the problems is that the coordinate system goes singular whenever we pitch, to plus or minus half pi (that means going straight up or straight down with our longitudinal axis). An example of that is if I first yaw and then pitch 90°, where the problem is, you see the orientation I've reached there. Alternatively, suppose I pitch and then roll, I've gotten to the same point. Now how can you tell me how much of that was roll and how much was yaw? It's indistinguishable. The coordinate system is singular there and it causes all kinds of problems whenever we try to work either at the straight-up-and-down orientation or even close to it. We'll see an example of that later on.

In a mechanical gyroscope where you have three gimballed bearings nested inside one another you can get to this condition called gimbal-lock where two of the axes are aligned. Once that happens the gyroscope is essentially useless because you can't get it out of that orientation. As a result, NASA has decided not to use Euler angles in any of its spacecraft. For instance, the Shuttle uses quaternians.

The next problem I'd like to show you is what happens when we try to take these Euler angle rates that were described in the Draft Standard and try to do a simple-minded forward integration. We take the three derivatives yaw, pitch, and roll and increment yaw, pitch, and roll by those derivatives to see what happens. To demonstrate this, Todd Shannon of BBN has put together a nice video clip three minutes long. First, let me say what it is we are going to be showing you. I know down at Disney World here you have Michael Jackson performing as Captain EO. Well, it's a little hard to compete with that but we're going to try. Today we're going to bring you "Captain Euler", who is going to fly an airplane exactly

according to the differential equations specified in the Draft Standard. Now, in addition to that, we are going to bring you "Captain Quaternian", who happens to be my very favorite super-hero. He's going to be flying his airplane according to the equations of Section 8 of my paper. For completeness, to round out our squadron of formation fliers that you are going to be seeing here, we've also thrown in "Captain Rotation Matrix", who uses the 3x3 rotation matrices that we've always used in SIMNET.

Video Presentation

Here we'll be initializing the three airplanes: Capt. Euler on the left, Capt. Quaternian, followed by Capt. Rotation Matrix. We create the airplanes, give them an initial attitude they're all pointing in the upward direction and then we tell them to take off. The maneuver that we have told them to do is a constant rate turn, one of these things I described before. It's just a loop. What we want it to do is make a closed path, a circle. Just go round and around. Now you see, in this case, all three of them are doing pretty well. Capt. Euler has fallen a little behind here but he's doing okay. The reason that this is working so well for him is that he is only turning through a single axis. Right now he's running his pitch axis around and around but his roll and his yaw are absolutely constant. As long as those two are constant, he does just fine.

Here's the second case. Now we are going to do the same experiment again, except we're initializing to a different initial attitude. The planes are going to pull their loop in a 45° plane instead of a vertical plane this time. The significant difference is that now Capt. Euler has to rotate around multiple-axes at the same time. He's not only got just a pitch rate that he had before but he's also got a roll rate and a yaw rate. One of the things that you see is that he is very quickly diverging from the behavior of the other guys who are just doing the loops as they should. The

problem is that we have forced him to integrate non-zero derivatives on multiple axes at the same time.

The next clip is going to be even a little more interesting because we are going to start them off at an attitude that is going to be very close to pointing straight up. Captain Euler is going to be close to the overhead position which I mentioned before, where his coordinate system is going to go singular. The result is that he is going to get very unhealthy roll and yaw rates, whereas Capt. Quaternian and Capt. Rotation Matrix are just doing their loop as usual.

End of Video Presentation

Now, that's an example of something that Rolando Rabines will speak to you about a little more this afternoon. Whenever we start working on something complicated it's very hard to predict all of the effects. Rapid prototyping is essential. We would never have guessed this particular thing without actually implementing the equations and seeing what happens. We should always plan on doing prototyping rather than just voting on which equation looks nicer.

Obviously, doing the simple-minded thing was wrong. Suppose we wanted to do it right so that the Euler angles can be used, and do sensible things like pulling those loops like the Quaternians and the Rotation Matrices were able to do. To do that we have to take a much more complicated approach. Here's the first equations for anyone who likes this kind of thing. You can take the Euler rates (these are the kind of things that were specified in the DIS Standard) by going through some nasty trig functions, and use them to calculate the omega vector. It is essentially the axis and the turning rate that we want to do the turning around. Then, given that we now know what the omega is (we want to maintain as a constant Omega during this maneuver) we can use that in a set of differential equations here to run forward the roll, pitch, and yaw

variables by again, using some nasty trig functions. In fact these are particularly nasty. I don't know if you can read it, but we are using Secant Thetas here in the top row. Now Secant Theta blows up to infinity whenever I get up to +/- 90°. I've got Tangent Theta here in the bottom row - same story. It goes to infinity whenever I go straight up/straight down. That was part of the effect that you were seeing in that last cut on Capt. Euler. We had a perfectly sensible Omega (it was about 1 radian per second) but once we push it through this equation and it gets multiplied by huge numbers here you get ridiculous rates on those two variables. Finally, if you were going this path, if you were trying to do Euler angles correctly, then you probably want to come up with a rotation matrix at the end in order to be able to do standard vector transformations to do the computer-image generation. All of this is on the order of 150 multiply/adds per frame time or per tick, assuming that trig functions take about the equivalent of 20 operations.

Now we want to propose a different approach here. Quaternians. Sir William Hamilton introduced the use of quaternians. This is the same guy who gave us Hamiltonian Mechanics and the ever-popular Hamilton-Jacobi Partial Differential Equation.

When we use quaternians what kind of math is involved? First, how do we represent a finite rotation with a quaternian? Finite rotation might be a rotation around some axis by a finite angle, something that we might do in a fifteenth of a second or a thirtieth of a second or a sixtieth of a second. All you have to do is determine what the angle of rotation Theta is that we are turning through at that time, take cosine of half that, and that is a scalar part, the first element of these four numbers. Take the sine of half that angle and multiply it by the axis of rotation that I mentioned before, and that gives you a vector part for the other three numbers out of the four. What do we have to do to each

frame in order to update our orientation as I keep ticking it along fifteen or thirty or sixty times a second? What's required is just to use this quaternian that we constructed for the finite rotation and do a quaternian multiply each time. What's a quaternian multiply? You have to multiply the scalar parts, have a vector dot product on the vector parts. The combination of those gives you the scalar part of the result. For the vector part of the result, you cross-multiply the scalar and vector parts of the two and you do a cross-product of the two vector parts of the quaternians. This is about 28 multiply/adds. When you're done you might want to go back to a 3x3 rotation matrix if you use that kind of a matrix to feed your computer-image generator. It takes another twenty operations to put it into the rotation matrix. We've got a total of about 48 multiply/adds per tick doing it this way. In other words, it's about a third the computation that was involved with the Euler angles - using the computation that I showed you before.

In summary, if we use quaternians we just do that quaternian multiply I just showed you. We don't have to do any numerical integration of a differential equation. There is no differential equation. It's just a multiplication there to find the orientation. Granted, it does involve vector dot product and vector cross-product. We have eliminated any errors that have come into numerical integration. Also, we've got no singularities in these frame update equations. Unlike the Euler equations where we had the infinities popping up every time we tried to go up or go down, here everything is finite. In fact, quaternians are normalized so that the sum of all the squares of the four elements is one (1). So every number that we are dealing with there is less than one (1). It's much less effort. So my recommendation is that the DIS Standard should use quaternians rather than Euler angles to represent the orientation. I believe that it's time that we eliminated Euler angles. They are old-fashioned, out dated technology of the 1750's. It's high time we adopted quaternians, moved up to the new, flashy, modern technology of the 1850's.

(Question) Well, I think the real issue here is not whether quaternians are necessarily a good way to compute mechanics, not necessarily whether the quaternians are a good way do dead-reckoning or not but whether quaternians are a good way to standardize on the contents of a protocol for communicating between simulators.

I think that in order to look at this you need to think about what kind of assumptions are going to go into choosing something for the purpose of being in a standard, as opposed to choosing something for the purpose of being in the design of a specific simulator. These are some assumptions, and like most assumptions, they are arguable. I'd rather go through them and accept them for the time being. Image generators work with Euler angles as inputs. There is perhaps one counter example where you can actually input the rotation matrix directly into the device, but the overall majority of image generators in the world accept orientation as some kind of Euler angle which they spread into a matrix which is usually implemented in some kind of hardware circuit in their card.

Low fidelity simulators are in a position to not do dead reckoning of orientation, for instance the SIMNET simulators don't. There is some band width expense to that if you want to have orientation portrayed correctly, but frankly, in a lot of the low fidelity simulators that we're talking about, as long as you can tell what direction the fighter plane is pointed in you don't really need a lot of accuracy as far as what its bank angle is or what its angle of attack is because your average tank on the ground can't make much use of that information. Another significant part of the simulators that we have to address here are devices which represent everything (tanks, airplanes) as a dot. I'm thinking of something like joint stars or some kind of naval track control system where everyone is viewed as a track and no one has the concept of what orientation of the airplane means.

On the other hand, high fidelity simulators are going to need to express the mechanics correctly and they are going to need things that move smoothly. They are going to need that in order to be able to perform aircraft relative maneuvers, air-to-air refueling, or simply flying in formation without getting confused. But on the other hand, I believe that those are the people who can best afford a few additional operations and I'm really talking here about a few operations that we'll get to in a second. So if I have to put somebody out, I'd rather put somebody out who was working with a high fidelity flight simulator because, frankly, I believe he is much more likely to have the computational margin to be able to afford a few extra instructions, to do a little bit more complicated process. That's what I'm talking about, switching to this little bit more complicated process.

As far as singularities and all of those kinds of things goes, I agree that the representation for angular velocity that's presented in the standard has some weaknesses. It does go singular at inconvenient times, like pointing north in certain places. On the other hand, I think that expressing angular velocity in terms of the center of mass of ownship and the center alignment of ownship means that you're not in situations where you're looking at 90° kinds of rotations very frequently because an airplane that's going through 90° rotations in the period that I'd be willing to dead reckon them over is probably out of control. So, although I agree that the sensitivity of Euler angles to that particular inflection point is a very significant mathematical problem as far as dead reckoning them over long periods of rotation, I don't believe that there are a lot of training situations where you're going to allow someone to dead reckon through three or four rolls. Because, frankly I don't think there are going to be very many pilots in a tactical situation that could hold a roll for three or four rolls. So what I'd like to look at is some alternatives, and as far as I can see, the alternatives are that you either transmit Euler angles or you transmit quaternians. There are two cases to

look at here, either you're a high fidelity simulator who needs to show the motion smoothly or you're a low fidelity guy who doesn't. If you transmit Euler angles, the high fidelity person, if he is going to do graceful, elegant manipulations, is not going to use that ugly omega equation because, that's just as hard as can be, at least 150 instructions. I think you're generous there. What he would do is simply convert the Euler angles to quaternians and use quaternians internally for all of his internal operations as well as for his internal dead reckoning, and then convert his quaternians back into Euler angles and stick them into his image generator to draw the picture. Now the cost to do that is that you have to do operation #1, converting the Euler angles to quaternians once every time you get a packet from somebody which is separated by whatever your mean time between dead reckoning updates are. That takes about 65 microseconds on a Motorola 68030 running at 25 megahertz with 68082 floating point co-processor if you do it as integers or it takes about 179 microseconds if you do it as a floating point. The cost of each cycle, each 15th of a second, each 3rd of a second, that kind of thing, is the cost of doing the second two operations, and that's about 170 microseconds on the same Motorola 68030. So you're looking at an expense that you're adding to the overhead of a high fidelity flight simulator that's 170 microseconds. Whereas, the overhead that you put on a low fidelity simulator, if you transmit Euler angles, is nothing. If you look at what happens if you transmit quaternians, the advantage to the high fidelity person is that he gets to use his quaternians as is. So the cost of operation #1 is zero. The cost of operation #2 and #3 is about the same. It's the quaternian multiplication thing that we just already talked about and can bring them back into Euler angles to stick them into this image generator which is the same 170 microseconds that it was before. So the delta cost that you save is 179 microsecond operation every mean time between rotates. Percentage wise that's very small if you consider that the number of iterations per second is probably 10 or 15 and the number of seconds between dead reckoning is maybe 2 or 3. You're

looking at something that's in the 1/30 to 1/50 of the time that he spends doing dead reckoning anyway. You save by having transmitted the information in quaternians instead of transmitting them in Euler angles.

On the other hand, you have the low fidelity guy who's getting away scott free for zero, if you transmit Euler angles. The situation if you transmit quaternians is different. The first thing he does is the same formulae that I was just talking about, he converts the quaternians into Euler angles which is simple, by the way. It's simpler than converting the Euler angles into quaternians, and then he doesn't dead reckon anyway. So his cost of two and three is still zero just like it was before, but his cost is 110 microseconds of floating point arithmetic every few dead reckoning seconds, which is infinite from a percentage point of view, but it's still not what I'd consider an obscene penalty in terms of microseconds.

I don't want to take the position that quaternians are intrinsically wrong. In fact, I like quaternians as an approach to handling a lot of these things. But I think that we ought to shift the computational burden in the interest of having a standard that would be applicable to a wide variety of machines. I'd rather shift the computational burden to the high fidelity people who have the computational horsepower to support it. Let's try, to let the low fidelity people and the people who have to handle a very large number of tracks, or the people who have to handle a very large number of targets with some low fidelity representation of them, beg off on the arithmetic involved in making that map. One man's opinion.

I agree with many of the comments that Randy just made, that any of these tradeoffs are possible. We can work it any way, we can transmit Euler angles or quaternians converted at the end back into the other flavor, if that's what we want to do. In fact there

are more dimensions than here. We don't just need to think about high fidelity and low fidelity, but we can do something tricky in between. For instance, I suggested applying this technique to aircraft, because they often do constant rate turns. It's not so applicable to ground vehicles. In the case of ground vehicles what we can do is look out through our computer image generator off the battle field and not use these more complicated techniques, this orientational dead reckoning for ground vehicles. We can use it for air vehicles. Better yet, maybe we should only use it for the ones that are close. If it's a tiny speck out there in the sky, who cares whether it's rotating a little bit or not. Don't bother. Only if it's up close enough for me to see it and resolve angles, then I could grind through all of these equations for each of the things in sight. So there are many different economies possible here. I think it's going to take a significant amount of work to try to work all those tradeoffs and come out with whatever is the best overall solution.

(Question) I'd just like to point out that I agree that the quaternians solve the singularity problem. But I'd like to point out that in most aircraft simulations we need Euler angles. The cockpit displays are driven by Euler angles, our sensors use Euler angles. So by and large we are going to compute them anyway. Now as far as the PDU, perhaps we ought to make a provision to pass them both. That might be a solution. The other point I'd like to make is about your assumption of a constant angular rate maneuver. I don't believe it is realistic for air combat maneuvering or air to ground weapon delivery. It might be appropriate for something like the airlines fly but if you've ever landed at National, coming up the Potomac River, it's not even applicable there.

(Question) Let me respond to the last point first. We actually did some experiments with real pilots flying at Fort Rucker in ground attack scenarios where they are after tanks and evading SAMs. We took the recording of all the packets that went on the

network during the experiments and then played it both ways. In one case we were pretending that we were just doing the current SIMNET kind of simple minded dead reckoning. Then we went back and played exactly the same data record over again, this time simulating the effects of having this kind of constant turn rate maneuvering in there, and we saw a reduction by a factor of three in the net packet rates. So even though it may not be perfect, it does yield, in the experiments, a significant improvement. I'm sorry, could you tell me your first point again?

(Question) Let me respond to that, I mean you talk about the reduction in packet rate, what about the change in performance or the change in the characteristic way that the pilots are flying the airplane? I think that's of more concern in a training situation. The first suggestion was to recognize that Euler angles will probably be needed to be computed anyway. They are available; perhaps the PDU should include them both.

Ok, let me respond to that one now. The use of Euler angles in your instrumentation is for the heading and the artificial horizons and so forth. That's not appropriate with these Euler angles. As I understand the DIS standard, these are Euler angles with respect to the world coordinate system, which means the Z axis is pointing up the North Pole of the earth. So if we take the angles with respect to those three coordinate axis, they have nothing to do with the local gravity coordinate system and it would be nonsense to try to use those very Euler angles and put them up on your display.

(Question) I don't agree, I think they are the same Euler angles.

(Question) Yeah, I think that was assumption #4, to switch to the Euler angles that he put up. I was referring in one of the assumptions that I made to switching to those ownship centered kind of Euler angles, instead of kinds that are currently in the

standard. I agree with you, that the ones that are in the standard need to be changed, but I think that you could just as easily switch to the ones that, as he said, you need anyway.

Let me press on that a bit, what's the reference coordinate system when you do that?

(Question) The center line of ownship and north, not through the axis that the wheels, they'll point left or right whichever one....

That's the platform coordinate system. What's the world coordinate system that we're transforming to the platform coordinate system?

(Question) Geocentric, earth centered.

That's the point that I was just making. If I measure these angles with respect to the geocentric coordinate system, north pole, prime meridian and so on, they have nothing whatsoever to do with what he wants to put on his heading indicators.

(Question) But there's nothing that says that the coordinate system that you use to measure orientation has to be the same as the coordinate system that you use to measure orientational velocity. You could still transmit rotation of velocity to update orientation.

You're saying that the angles I transmit and the angle rates have nothing to do with each other? One is not the derivative of the other? Ok, you're going to have to write a white paper for that.

(Question) Yes, DARPA is interested in networking the simulators for the X31 aircraft. The X31 aircraft which flies at very high angles of attack, rolls about the velocity vector. It's not clear

that either one of these ways of describing the performance of the aircraft would be adequate to enable us to do that kind of thing with the network.

What's your concern?

(Question) Just what are we talking about here? Are you talking about Euler angles? Is the velocity vector necessarily tied at the water-line of the aircraft? How do you compensate for that? How do you make sure that you can display it properly? It seems without having delved into the mathematics as much as you two have that the system that you're looking at would not allow us to depict that, or to fly those airplanes in that kind of a simulator very well.

Ok, you brought up the topic of the velocity vector. We haven't mentioned the velocity vector at all today. It's completely independent of what we're doing with orientation. The intention here is that we were just talking about representing orientation, not talking about representing translational motion at all. We have to do that of course, but it's completely independent of all these issues that we raised.

In my proposal the quaternians represent angular transformation to get from the geocentric world coordinate system to a local platform coordinate system of forward, right, and down.

(Question) What do you do when the omega vector is not constant?

During the time that we are doing this dead reckoning the omega vector is a constant, its just an axis and a rate around it. If your velocity changes, then you send another packet and say, hey, now I'm doing something else. Forget that roll, now I'm turning left. This is only between packets that we use this extrapolation technique.

(Question) You may have just stolen my thunder there in the very last thing that you said. All these people, who apparently are pilots, are discussing what airplanes do in so many seconds or don't do. The one thing that I wanted to clear up, and you may have just done it, is that when you take an airplane like an F-15 or F-16 that's maybe doing 500 knots, in full afterburner, that decides to go this direction from it's present heading of straight ahead, goes idle with speed brakes and pulls 9 G's, there is nothing constant in the rates of acceleration or deceleration. I just wanted to make damn sure that point is clear, that there is nothing constant in the air-to-air high performance. If you're at Fort Rucker and had air force pilots, you had real pilots. But if you're using the A-10s you didn't have real airplanes. You had something only slightly faster than a tank. The point is that if you're using dead reckoning, you're now using the new decayed or increased angular velocity or acceleration rate which you must bring in there. Then I agree with you. But if you're holding something constant over a point of several seconds, in a high performance fighter, that would be a totally wrong. But I think you said till the next update is made, and that may clear it up.

Yes, but the update rate is not fixed at all. We send an update whenever our dead reckoning model exceeds a threshold error from what the real aircraft is doing. So in violent maneuvering we may be sending out packets at the rate of three, four, or five a second because it's changing what it's doing all the time. On the other hand, if it does start doing constant rate turns it sends one out. It doesn't have to report anything because it's not changing its maneuver very often. Only when it changes its maneuver does it send out a new one.

(Question) I'd like to add to that in relation to high performance aircraft, also high performance weapon systems, and the comments earlier this morning about doing any sensory work. I think before you go on, you ought to rethink what was just done in this

discussion and go back to the paper, the presentation before. That is, what's the real direction for this networking activity? I think Tom Hoog has outlined a position that says, you better understand where you want to get to and the road map you're going to use, before you start asking if you want to use quaternians or if you want to use Euler angles. If you don't need them, obviously there are equations and cost tradeoffs that you can show and support that say don't use them. But, if in fact you're going to need that data you have no other alternative.

As I said before, this is an area that's going to need further study, further tradeoffs. Randy and Ray Fitzgerald are members of my committee and work on these problems and come up with another white paper in time for the December/January draft standard. We will try to get that firmed up a little more.

(Question) This is really a repeat of this gentleman, but I just want to make it clear the assumption that you're making is that omega is fixed and therefore the unit vector is constant. When you get a new position, or a new quaternian, what are you planning to do, just replace it? I'm just asking based on your experience because then you're going to get a jump or some jerk.

Yes, that's another good question. We have suffered from that a little bit in a thing called Stealth vehicle in SIMNET. That's a large screen display that lets a viewer move invisibly around the battlefield and go anywhere to see what was happening. One of the modes we have in that Stealth vehicle is a tether so that we can essentially throw a tow hook onto the back of any other platform on the battlefield and have it tow us around, and we can watch in detail what it's doing. Now, as that vehicle moves around the battlefield it sends out (from time to time, whenever an error threshold is exceed) a new update packet and if we just take that packet and suddenly update the state, The user says "I was here doing this and now, oops, I'm suddenly here doing that" The

difference here is whatever the threshold was that was set earlier requiring the updates. We see a jerky appearance in that Stealth Vehicle. In order to fix the appearance of the Stealth Vehicle we've done about a half second smoothing, so that instead of having the thing suddenly leap by a meter, or two meters, or whatever we set for the threshold, we take that step linearly over a half second so that we don't get disturbing jumps or twitches in the behavior that we're watching through the Stealth vehicle. Now ordinarily, on the battlefield, most of the other vehicles, certainly enemy vehicles, are far enough away that you're not going to see that effect. It's not going to bother you for reasonable size thresholds. But in the case of our Stealth it was disturbing, so we did put that smoothing effect in.

(Question) So you may need to add that to your PDU, right?

Well, its beauty is that it's entirely in the eye of the beholder. That's all processing done at the receive vehicle in order to improve the psychological effects of the appearance. It doesn't affect what goes onto the network or the source vehicle at all. So think of it as a trick in the final CIG to smooth out disturbing jerks.

(Question) I might make just a few quick points. When we went through this tradeoff of Euler angles verses quaternians, there was no intention on our part to diminish the effect of quaternians because they're used in almost all high fidelity flight simulators. What we looked at was where it made sense for the computation to be performed. Some of the data we had was from some exercises we had run on our little F-16 simulators with air force pilots, which very infrequently during air combat showed constant omegas. The other thing was that when we were thinking about this we thought that things like roll stabilized missiles or commercial airplanes could have the singularity at a point that would never, in reality, occur. So we wanted to give as much flexibility, if you will, to

the individual simulator house to deal with the angles as they saw fit. That was the purpose of picking Euler angles verses quaternians.

As Randy pointed out, we can do it either way. We can send either one on the network, we can convert it at the other end. All we have to worry about is what it costs. We should do careful tradeoffs there, particularly with real data. That is another point. It's good to do this with real data, not with just paper and pencil.

"Bit Encoded Attributes in Distributive Interactive Simulation: Why They are a Bad Approach and How to Fix Them," Randy Saunders & Mike Robkins, Hughes Sim. Systems

We'll discuss why we don't think bit encoded attributes for describing entities is the right approach for all the information that will be simulated. I will show some problem examples that were brought up from bit encoding information, and give some possible solutions and alternate methods for sending the information that don't depend so heavily on bit encoding.

First, some bad examples. The standard, the draft standard as currently defined, leaves room for only 32 possible countries. The standard left out some of the largest armies in the world and some potential adversaries and allies like Israel, North and South Korea, India and Pakistan, all of the Southeast Asia Treaty organizations except for the USA, the PLO, most of the Middle Eastern countries, including Kuwait which shows how quickly a standard can be outdated. The draft standard for bit encoding information also required that all possible weapons, sensors, platforms, vehicles, munitions and configurations be predefined and hard-coded into the standard. Jane's lists at least 18 variants of the MIG-21 (I had an old book, and there may be more) and five variants of the F-15s. It's conceivable that future simulators will want to make a distinction between different variants of the MIG-21, and this would require updating the standard. Other problems occurred with articulated parts. B-2 split spoilerons, B-2 engine inlet doors, V-22 rotating nacelles, and LHX NOTAR louvers are all articulated parts that have appeared, at least to the public, within the last few years. None of these were defined in the standard. This is a quick list I came up with of currently undefined articulated parts. I don't know if they belong to any simulators yet, but its conceivable that someone would want to include them. As you can see, it's difficult to define even the current requirements for articulated parts and its going to be

impossible to predict future ones. All these have appeared within five years. Five, ten, twenty years from now there will no doubt be many more articulated parts we haven't even conceived of yet.

There are several disadvantages of bit encoding in the draft standard. Bit encoding is not appropriate for static or complex information. Bit encoding reduces the data size at the expense of flexibility, and since static information only needs to be sent infrequently there is no need to bit encode it in order to reduce its band width. Bit encoding also requires defining size, form and content of all the data. This is a difficult task in itself. Once you define this data you put a limit on the size and the amount of data that you can transport. Like the 32 countries, maybe later you'll need 64 countries and you'll have to change your bit encoding standard. Changing the bit encoding standard makes forward and backward compatibility between simulators highly unlikely. If you have to change the way you format your bit encoded data you're going to have to change the software on all you're old simulators. Either that, or add the information onto the end and change your parsing. Bit encoding does not adequately define articulated parts. Bit encoding is only appropriate for easily defined high frequency information.

Our solution to this problem is to break up the appearance information of the entities into three PDUs: a Definition PDU which contains only static information, an Articulated Parts Definition PDU which defines and lists all the articulated parts that will be simulated, and an Appearance PDU which contains position, attitude, acceleration and the position of all the articulated parts. This information does change very rapidly and is encoded to reduce its band width. The other information is mostly static and there is no need to encode it.

An example of the Definition PDU is a tank. The first field is a change flag that says this string has not changed, it's

currently in the land domain, it's a tank, it belongs to the US army, it's an M1, the version of the M1 is A1, it has a voice radio, type SINCGARS, it has a marking, 1,2,3,4 and 5. Another example is for a ship. Suppose this entity is new so the change flag is true. It belongs to the Sea domain, the US Navy, its a guided missile cruiser, type CG-54, it has two types of sonar, three types of radar, and it has a marking in tedium.

The Definition PDU contains static information. It defines the entity, its capabilities, armament and markings. It would be transmitted every three to five minutes, or after a query. Randy Saunders will talk about our ideas about query protocols later in the afternoon. A change flag marks the data as changed or new so that simulators only have to parse out the information that they need to. It consists of hierarchical text strings made up of key words. The key words can be parsed out to any desired level. Tanks only need to go through two or three words to see they have a sub. They could ignore the rest of the data or ignore the whole string. New technology can be incorporated just by adding vocabulary to the standard. If someone comes up with a new sonar you just have to list it as a vocabulary word in the standard and people can add it to their strings. Even entirely new capability fields can be added without modifying old simulators. For instance, in this F-16 example it has a radar and IFF targeting capabilities and ECM capability. If someone, for example, invents an IR jammer, that capability could be added by just defining IR jammer as a key word, defining the various types of brands of IR jammers, and adding them to the string. In addition, old simulators can ignore or redefine key words that they don't understand, that have been added since they were built. This doesn't cost them any fidelity. If an F-16 simulator doesn't know about jamming pods and it sees the words ECM it will ignore them because it doesn't understand them. It will go on information that it does understand. In addition, station/pylon/launcher/weapon combinations can be defined this way. For example, in this F-16,

at station four there is a weapon pylon, a triple ejection rack, two Mark 82 bombs and an empty hard point. This sort of syntax could also be used to describe ship to ship missiles on launchers or any other attachments to any vehicle. This sort of syntax is applicable to high and low fidelity simulators. For instance, a tank simulator may not be able to distinguish between various versions of the F-16. It may not be able to distinguish between various fighters so it would only parse the string out to here and would note a display of fighter. A high fidelity aircraft simulator might actually be concerned about the locations of bombs on pylons, but that information is also contained in there and can be parsed out. Distinguished network entities or application gateways can provide additional data to support low fidelity simulators in a high fidelity environment. For instance, if you have a low fidelity tank simulator making this string to be broadcast to other simulators and it doesn't know what kind of radio it has, a network entity can add the voice radio SINCGAR string onto that when it's broadcast to other simulators.

This is our proposed syntax with the Definition PDU. It's a string composed entirely of predefined key words, each word separated by a delimiter. In this example I've used a period, although almost any string will work. Words describing entire capabilities are enclosed by () to make parsing easier. In the example, the sonar was (sonar) and then the types of sonar on the ship. This is the hierarchy; the important basic information is first and detailed high fidelity information is last.

The Appearance PDU: This contains all the dynamic contents of the appearance PDU like location, velocity, acceleration, orientations, angular velocity, whatever format we decide on. Other bit encoded appearance information that won't change with technology, like wakes on ships, or smoking tanks, are added on the articulated parts record. For this, a new format is needed.

This is how we propose to define our articulated parts. Articulated parts are broken up into two pieces, one a static data PDU. This is sent infrequently or after a query. All this is a list of the articulated parts that the simulated entity defines. An aircraft entity, for example, might have ten articulated parts: a left flap, right flap, rudder, landing gear, tail hooks, etc. This string would be parsed out infrequently, every three to five minutes or after a query. The dynamic data is sent frequently so it's compact in order to reduce the band width. We use integer values for the articulated part positions, the units predefined by the standard and the vocabulary. In this example, the number 1,234 would represent the position of the left flap, 5,000 would represent the position of the right flap, etc. This string would be broadcast at high frequencies, but at the smallest possible size.

The receiving entities can identify from the static string which fields of the dynamic field they need. New fields and new articulated parts can be added without upgrading existing simulators. All they have to do is add new vocabulary to the standard.

To sum up we use three PDUs to define the entity appearance and capabilities: an Entity Definition PDU containing mostly static or complex information, an Articulated Parts Definition PDU which only lists the articulated parts that will be simulated, and an Appearance PDU which contains all the rapidly changing information such as position velocity and the position of articulated parts. This setup is flexible where it's appropriate to define entities and compact to minimize the band width of rapidly changing data.

(Question) I have a long question. I certainly agree with the sense of defining, ranging entity type codes in some sort of hierarchy. It allows the simulator to be able to understand something about an entity without having to know everything about

all possible entity type codes. There are a lot of different ways to arrange that hierarchy and we ought to, in choosing a particular way, consider how that hierarchy is going to be used; how those entity type codes are going to be used. It seems to me that one application of the hierarchy is to allow image generators to be able to draw something representative of a vehicle without having to know how to draw all the minute details of a particular type of vehicle. So a simulator should be able to recognize, for example, that something is an attack helicopter flying in the air, without having to know all the details about AH-64s or something like that. So the top couple levels of the hierarchy, as your hierarchy has them now, ought to include what environment the vehicles operate in, what class of vehicle, and what sort of function the vehicle is meant to perform. Now, I see you've changed you're hierarchy since you wrote your paper, and this sort of points, I think, to how difficult it is to come up with a good hierarchy. It's an area in which we ought to be discussing and considering the various tradeoffs and the needs for the different levels of the hierarchy. I'm not certain I've seen the requirements stated for having the hierarchy go to such a fine level of detail such as yours, down to the level of markings capability, sub-version numbers and so forth. I think that needs some clarification. Why do we need to include that sort of information in what is sent about on the network? That's one aspect of my question. Let me set aside the hierarchy for now and draw a line, and also mention briefly the encoding scheme that you've proposed. You say that strings are more flexible than a bit coding representation because, first of all, unlimited length strings no longer impose some strict limit on how many different values you could represent, and secondly the strings can be more extensible in the future. Well, I think for most of the fields that we need to consider, like the countries for example, we can come up with fixed upper limits on how many values we will ever need to represent. In the case of countries sure perhaps 32 isn't going to be enough but would 16,768 be enough, perhaps. We could draw the limit there if that seems like a

sensible thing to do. Bit fields shouldn't be excluded simply on the basis of not being capable of representing large enough ranges of values. In terms of flexibility, as you point out, you can make up new sorts of strings and add them to the standard list of new string names and include them in the standard without disrupting existing implementations. It seems to me one can do the same thing with bit field values. Define new bit field values in the lower layers of the hierarchy, simulators that understand the higher levels of the hierarchy will continue to be able to recognize these new values. My question then is, what do you think about all that?

Excellent question. The examples I used were only examples. They were meant to show the capabilities of this sort of format, not how far, or how detailed we think it is appropriate or will be appropriate. That's, of course, up to the working group. The other question was can we use bit encoding as sort of a string instead of actual words, and that is quite possible. The other point I wanted to make though, is that information should be sent infrequently and if it's sent infrequently then there is no reason to bit encode it.

I think that you clearly take the approach where you just assign all the words in your vocabulary a number. For example, A is #1 and Apple is #2 or whatever they turn out to be. You could just send those numbers and that would certainly save you some letters. But in the country example that you talked about, I'm sure that two bytes worth of countries would be plenty of countries. But Czechoslovakia only has 13 letters in it, and so I don't know that you're making your programmer's life so much easier by making them look through this book to find out that Czechoslovakia is word 496. It's worth it in comparison to the additional 11 bytes of transmission time that you're saving as long as you accept the concept that these are things that you transmit very, very infrequently, because what country, whose army your tank is on should be something which is a slowly changing

parameter. I think, though, the benefit is in being able, for example, if you want to know what country it is, to type the name of the country. And if you want to know how to spell Czechoslovakia you look that up in the dictionary which you already had to buy, as opposed to always having to go to a book where you could put the same words in and make the programmers look up the numbers. This is a decision on how you want to shape the implementation work of the programmers. Do you want to make it so that users have to look it up in the book and type 472? No, they're probably going to have to pick them out of a menu so that means somebody has to type your document in away that means, when the guy picks Czechoslovakia off the countries menu on this MacIntosh that you put 472 into this field. It's not impossible, it's not a technical challenge, but it's another one of those areas where I think that we can make our standard more easily accepted if we take the easy things in the places where we can afford to take the easy things, and we save the hard things for the few areas where the performance is so significant that we really do have to have them.

(Question) I think you're confusing a user interface with a protocol. The main purpose of these PDUs will be for simulators to interpret them and be able to draw images and simulate things on the basis of them. So therefore, representing them in the form that's most easily computed with, manipulated by a computer and communicated seems to me the most important. And providing some form that can be conveniently printed out on a teletype or something, is definitely secondary.

Definitely secondary.

(Question) That argues for the bit representation.

(Question) However, if we're writing in Ada we're not going to have a code that says 1,2,3,4, you're going to have a name on it,

and if you have enumerated type it could just as easily say Czechoslovakia as 1,2,3,4. Who cares? That's a real detail, I think the point is well taken, what that actual string says. No one cares; our code doesn't care. It could mean anything.

Well, your code doesn't care as long as you accept that your Ada compiler is going to convert Czechoslovakia into the same number that my Ada compiler is. Since the conversion of enumerated types into constants is a Section 14 thing where you are allowed to do it differently in every Ada compiler, I'm not so sure that I'm going to trust Ada compilers to do the dictionary. I'd pick Art Pope over an Ada compiler. You're right, there are some decisions that have to be made about how compact is compact enough and not to steal the thunder of some people that are coming up later I really think that empirical studies are the way we need to make a lot of those decisions. We really bring things up mostly to surface the issues in the hope that the people at IST will invest some calories, and do some studies to try and come up with some more scientific basis for making these decisions than the ad hoc ones that we propose.

(Question) This discussion is beginning to sound a lot like we ought to call Tom Hoog up here again, because a lot of the points which are being brought up have to do with why are we playing and what are we playing. Number 1, if you're going to have a simulation of war games, it's not a free-for-all. You've only got a certain number of players and there has to be an exercise controller that says who is going to play and what's their starting configuration. You have in there a change flag, you also said we're going to update it every three to five seconds. Why? If you update it unchanged, that's all you need. If somebody's going to join, the exercise controller is either going to let him or not and the exercise controller better have him identified as to what his capabilities are when he joined. Now you really make it complex if you have giveaway bombs and rockets on airplanes part of his basic

configuration. I can see giving him a capability 6 stations, or 8 stations or whatever, but it's probably better treated in the articulated type thing. That's a real good one there to break that out. Because the giveaway stores, the articulations are clearly things which are weak in the DIS PDUs. But we really have to start rising above some of the old things that you have to keep updating because you don't know who is on the net or why he's there. If you're going to talk about simulations or mission rehearsal (what I'm more interested in) you have to play with the set piece that you're playing with. Why do you want to know if the country Czechoslovakia is the place where this airplane came from? The only valid use is if Czechoslovakia uses, or their pilots use different tactics than a Russian guy flying the same airplane, that's all. You don't have to spell it out; you don't have to see what country it is. It's to show if you're letting a machine fly a SAFOR what's it going to do. So there are some good things in there, but I don't think that it's going to be hung up on a string versus bit configuration. It's going to be done on architecture and what are we really trying to do.

<u>"The ACME Radar PDU: An Alternative Approach to Emissions," Alan Oatman, BBN</u>

I'm Alan Oatman with BBN and I'm here to discuss the ACME Radar PDU. I'll freely admit that I'm one of the wallflowers as referred to this morning, as someone who was looking into this area for a task that I've been assigned to and hadn't really put any ideas down on paper until I saw the draft standard. I felt there were some things that just could not be left out even on a first cut. I'm not here to try to say that I've got an answer to everybody's problem or that it will last long. I have a particular task to get done in the next few months and I'm going to present some ideas of how I plan to do my job. First, I'd like to thank Brian for eloquently stating the problem that radar is a big problem. I definitely think that's the case, and I think it's much larger than one person or even a body this size can solve in a short amount of time. My intent is to produce a PDU that is adequate for the short term, but to do that intelligently to allow some limited growth and sophistication in the near future. The emphasis, I think we need to remember on emissions, is not necessarily on the system that's doing emitting. It's more on the systems that exist, that can sense that emission. It's a little bit different way to look at a radar simulation than is typical, particularly an aircraft simulator. Now we really need to look at what that radar looks like from the viewpoint of a radar warning receiver or other passive sensors.

Here is a little bit of background on what I'm working on. You've heard the term ACME. That's really not a network that you buy from Warner Brothers, it stands for Air Crew Combat Mission Enhancement. The protocol has also been known as SIMNET Air Force. What we're trying to do is put together several existing simulators and also integrate them with some new equipment. We are going to have an F-16 high fidelity in the near future. In the very short term we're going to have an F15 medium fidelity. Currently we have

several F-16 low fidelity devices. Just as kind of a side note, the F-16 low fidelity devices do have their own protocol that has already been implemented, and we will probably be doing a translation between that protocol and SIMNET, kind of going along the lines of what we heard this morning. There is also a new device coming on-line that is a threat generator that's also going to impact, dramatically, some of the issues that will be discussed today and tomorrow. As well as this, we will have SIMNET devices, the Plan View Display, a network operations and maintenance station, a data logger for record/playback, and in the future, hopefully, a gateway so that we can talk with other sites.

I'm advocating the adoption of the ACME Radar PDU in the DIS Standard. I'm not a hard guy to work with, if you don't want to adopt it that's fine. Make some inclusions into the existing one. That's really the point. The information needs to be in there. I don't care about the name, who authored it or anything else. We need some more information in this PDU.

The existing Emitter PDU proposal identifies entity identifier, entity type, time of emission, emitter location, number of emitters, data base number, and data base access information. It was certainly my first idea that we should not do anything like a data base for emissions, that we should be putting parametric information onto the network. I quickly walked away from that idea after starting to investigate what modern radars are capable of doing. I probably am convinced that the network could handle the band width, but I absolutely know that the nodes I am going to be working with cannot handle that processing to take in all this physical information. They really want to know what type of a system it is, what mode it's in and I'm going to look up on a truth table to see how well my sensor equipment can detect that emission. So, I do agree with the idea here for the use of a database. I would like to see some other information included.

I work in an air force laboratory and I know very little about ships except they are usually wet someplace and tanks, I'm just not that concerned about. To me they are just a nice little target. Fire control radar, however, is something I'm interested in. The first case, fire control radar, these systems are typified by systems that are shorter in range, multiple modes, normally directly linked to weapons systems (that's an important point), and have finite target capability. As opposed to these types of systems there is another one that I would call long range search. These are typically very long range, normally a single mode, maybe two. Normally they're not directly linked to a weapons system and they have nearly infinite target capability. Originally, when I started out on this data base, I decided that what I wanted to do was present a target list for the fire control radar where he would look out over the situation and say, "ok, I've got this great radar simulation running. I'm going to hand view a list of everybody I'm detecting." That's really clean, I really like that, and I thought long range search, if I've got an AWACS up there with 10,000 targets in its list I really don't want to have to put that on the network, so maybe what I'll do there is just include directionality information. What direction am I emitting? Discussions with other people in our subgroup or sub-subgroup quickly point out the fact that there may be passive sensors out there that you're not aware of. And therefore, just providing a target list is not going to be adequate. I may not be aware of their existence and yet they are going to be very interested, possibly, in my emissions. Therefore the two ideas seem to merge into a single PDU. What I'm trying to do now is see if we can agree on one method of presenting systems that would typify both of these.

I've come up with an ACME Radar PDU. Very similar to the DIS, with a little bit different terminology. I've got an identifier, location, time of emission (that's an asterisk indicating that has been added since the paper was submitted), and a number of emitters. Based on the number of emitters I will identify the

radar system, a gross radar mode, and specific data (specific data right now I don't have a use for. I included 64 bits to include subtype of specialized information. It could be polarization, staggered PRF, etc). Here I'm including directionality information. It's important to note that this directionality information applies to the radar scan. This is not beam information; we're not tracing a beam going back and forth through a raster scan. We're trying to identify what the current volume of coverage currently set up is. Therefore, if you care, you can build up this volume and determine whether or not if you're in it. Also, there is an element for power. Now down here I've got what I would call a convenience feature. It's not required, however it will make some jobs, especially air to air, much easier. This is a number illuminated, and based on that number I will identify the vehicle that's illuminated. I also have another word out there for particular radar data applicable to that vehicle. This will allow the situation where you're flying along and your radar warning receiver needs to be stimulated because someone is laying you up, maybe 2,500 feet on your tail. I don't want to have to build up maybe another 100 player's volumes and determine which of those volumes I'm inside. It would be very nice if someone tapped me on the shoulder and said, "by the way I've got you in single target track and I'm about to launch a weapon." That is cheating. This does assume that people are going to want to play fair on the network. I personally don't believe we can get a protocol that does not make that assumption. However, this information is being calculated in several radar simulations. Why not make the opportunity to pass that information along? If you've got the information, pass it along. Low fidelity devices could play in this game. Otherwise they will not be able to. Building up volumes is going to be a very large task. I would imagine that the task I'm going to go through at first will use only this area. This will be there for growth or for passive sensors that may want to build that up if they would like to.

The problem with databasing is that this directional information normally, my experience shows, cannot be encoded. Many fighter aircraft have the ability to continuously move that volume around. You can't just try to set up some bits and say, "ok, I've got it in this wide a scan and it's put in this place". That can be slewed around by the pilot, als by maneuvering the aircraft. I think it has to be explicitly stated. Power, on the other hand, is still under some research. I'm going to fall back on that one a lot. Like I said I don't have all the answers, this is where I'm heading. Power is one that I'm still kind of working through. Do we need to include that explicitly or should that be encoded in some type of a mode? Also, not indicated here is a revisit time. That may or may not be an important parameter regarding that directionality information. Can revisit times be modified continuously? I can think of some cases where they are. Again, that's an area where I need to discuss with more people who are here who have a better feel and understanding of the radar sensors that exist today and may be coming around the corner.

I think that this plan maximizes the flexibility of the PDU and will allow, at least at first, the elements that are important, especially in air-to-air, to be included. I believe that the models I've presented are applicable to SAM sites. Typically they will have a long range search, handoff to an acquisition, handoff to a track that will actually do the lock on and launch a missile based on the guidance. I believe that it also will apply to many surface ships that do exist. I would like to stress the fact that this is under development and I'm not trying to force an answer down anybody's throat. I've done some work with this group. I'm working in the radar sub-subgroup. I'm also in contact with people in industry and within the air force, trying to get some help, trying to get my hands around this problem. It's a large problem, but within the next six months it's going to be implemented. This will probably not be the final solution but I think HRL is an excellent opportunity. People here have been talking about

prototype validation. HRL is an opportunity for me to do that. What I would like from this body is the opportunity to do that. I've got to get this job going. If we stay with the existing emitter PDU it's inadequate for my task. I can't use it. I need directionality information included or a target list, preferably both. I'd like that opportunity to exercise some of the ideas here at DIS.

As I said I'm not going to solve everyone's problem, such as radar jamming, deception and noise. I'm aware of their existence, and I've really put very little effort into trying to answer those concerns within this PDU. Some other problems: IFF interrogation, bistatic radar, security, cooperation with weapon handover. Well, that's a bad word right now, that's not even included in the DIS. Also, JTIDS, boy that's a big problem. I think everyone sees occulting as a problem all over the place.

The proposed DIS Emitter PDU is inadequate, at least for my task. The proposed Radar PDU will resolve many of the immediate inadequacies, and I think it should be included in the standard, at least as an interim measure. I'm afraid I have to lean back on the "much more research is required." Emissions are a very important part of any Service's interested in performing a mission. It's something that has to be answered fairly objectively and well defined. Again, I've got a short fuse. I can't wait for two years until we determine what database we're going to use, etc. This is what I'm going to implement. I will field some questions. I forgot, I did want to mention that I'm not working in a vacuum, I'm working with people in the Air Force, personnel from McDonnell Douglas, General Dynamics, Analysis and Technology, and CAE link. I'd like to invite some assistance from some other people that I recognize as experts in the field, from Hughes, AAI, and Westinghouse. These types of people could really help us out as far as determining what the important parameters are in emissions today and possibly the near future. Obviously we can't predict

what's going to come around the corner, but we can intelligently see what's coming right up. Any questions?

(Question) Yes, my name is Laurie Miller, I applaud you for your recommendation. Tri-Service currently has a panel as well as SAB that is looking exactly at low fidelity, medium fidelity and high fidelity radar parameters. Currently, if you just look at medium it's 127 parameters just to describe any radar. It looks to me like we've got a very long way to go. My recommendation is there is a new radar panel on Tri-Service that you might also want to get the parameters and formats from.

Thank you, I'll try to get that information. Any other comments?

(Question) Dick Gagan, when you were describing the applicability to SAM systems, I could identify what you were saying with the characteristics of the Army's HAWK air defense missile system but not with the Patriot which uses a phased array radar. I wonder, have you considered phased array radar characteristics such as found in Patriot and the Navy's AEGIS system?

No I haven't explicitly. My explicit interest right now is aimed at fighter aircraft and trying to show some awareness of other radar systems in existence, particularly in AWACS. I am working on it from an Air Force spin, if you will. But I would like to do this to try to help out this body. They seem to be two tasks that are very much aligned and I'd like to see how much work I can apply towards both. Anything else? I'll interpret that as overwhelming support then.

"Seven Critical Technical Issues In the Draft Military Standard for Distributed Interactive Simulations," Richard Schaffer, Alan Dickens, Brian Vaughn & Art Pope, BBN

Good afternoon. Today I'd like to discuss the position paper I wrote with several co-authors at BBN. Our goal here is a very challenging one. Defining the application layer for DIS seeks to achieve the goal, at least ultimately, of supporting the full complexity of the modern battlefield. This is certainly a large task. In meeting the first step of this task, the final draft release of the protocol, we've identified seven major issues that we think need to be resolved before this final draft is released. I will address those seven issues in this talk. For detailed line-by-line comments see the other position paper cited in the slide.

Finally, at the conclusion of my talk, I'll make a few suggestions about how we can resolve some of these issues by December.

Here are the issues:

1) The scope and communications requirements (need to be better defined)
2) Dead reckoning
3) Issues relating to articulated parts
4) Restricted data representation
5) Dynamic thresholds
6) Weapons fire
7) Data representation issues

The Scope and Communications Requirements

We believe that the scope of the protocol and the communications requirements need to be better defined. There are many implicit design considerations in the design and these need to be made explicit. For example,

- Functions to be performed by the DIS System and the goals and constraints in its design

- The architectural model of the DIS System (which really means defining the components that are involved) and the interfaces between them

- The communication services upon which this protocol is based. (For example, multicasting and which kinds of PDUs are issued using which kinds of services)

Dead Reckoning

The next issue, which is a large one and a central one, is dead reckoning. What is dead reckoning? It's the procedure upon which a simulator calculates the state of remote vehicles in the intervals between the receipt of Entity Appearance PDUs. The draft standard proposes that this be done in the following manner: first position and world coordinates are dead reckoned based on velocity, world coordinates and acceleration in world coordinates. Orientation is dead reckoned based on Euler angle rates and articulated parts are not dead reckoned.

Here is the first issue we see here. It's often addressed as the coordinate system for the velocity and acceleration vector. The actual coordinate system doesn't matter much. The vector is the same vector no matter how it's represented. However, what is important is how this vector is dead reckoned when orientation changes. What's implicit, but not stated in the protocol, is that these vectors stay constant in world coordinates. However, you can do much better than that. For example, take acceleration. Whenever a parameter stays nearly constant, like acceleration, you can avoid sending PDUs for a period of time. If you express acceleration in world coordinates, then linear accelerations result in a reduction in packet rate. However, if you express

acceleration in platform coordinates then constant rate turns also result in a reduced packet rate. Also, let us consider velocity. A vehicle in the turn has its velocity changing very rapidly in world coordinates, however, it is nearly constant in platform coordinates except for any changes in speed as it turns.

The second concern is the limited flexibility of the dead reckoning policy. While what I just described handles almost all ordinary vehicles, certain vehicles, like ballistics objects that are tumbling, it handles very poorly. For example, because it's a ballistic object its acceleration is very well defined in world coordinates. It's just the acceleration of gravity. However, because it's tumbling that acceleration vector varies wildly in platform coordinates. So we recommend that there be a mechanism by which a dead reckoning class be defined in the protocol.

Third, dead reckoning of articulated parts is not specified in the protocol. However, we found that in the SIMNET experience with ground vehicles that have turrets the major reason for issuing PDUs is that the dead reckoning thresholds for rotation have been exceeded. That is, if you wanted to reduce the network traffic in the large combined exercise, which by definition would have a large number of ground vehicles, you might be better off using dead reckoning of articulated parts rather than high order dead reckoning for the relatively smaller number of aircraft.

The next point, I think, has been adequately addressed in Jerry Burchfiel's talk.

The final point here with dead reckoning is that it's sometimes suggested that high order extrapolation is optional on the part of the receiver. There is a suggestion of that in the statement which uses the word 'may' in the rationale document. I'd like to present an example which I hope will make very clear that this is not an acceptable approach. We have two cases here, in

this cartoon. First, the ordinary case where both the sender and receiver use the full high order extrapolation in the top panel. In the second case the receiver is ignoring the acceleration data sent by the sender. Here is a situation. We're concerned with this missile simulator. It's launched at a certain time, flies out, and is trying to hit the aircraft. We can view this from anywhere in the world considering the point of view of the aircraft. In the top panel case the missile starts at time T=0 and sends an initial vehicle appearance PDU. Its position hasn't changed. Its velocity is at zero at the moment of ignition, but its acceleration is very high, many G's. In the top panel case the missile flies up, and the pilot has a chance to detect it and take some kind of countermeasure and perhaps evade the missile.

Consider the second case, because the receiving simulator is ignoring the high order rate and it will not see the missile move until the missile simulator sends a PDU for some other reason. Let's assume here it takes the SIMNET case of a five second time out before you must send a new PDU. In that case, the aircraft simulator will see the missile suddenly teleport from its launcher to about half a click away in the case of a 4 G acceleration at a high velocity.

So, what's the moral of the story? First, notice in this past case, the sender's internal state was the same. It had the same full detailed model, it thought it was flying at the missile. Second, is the key point. Dead reckoning is based on a contract issued by the sender. This contract says if you follow the dead reckoning logarithm with all the data I've passed you, at any point in time I'm guaranteed to be within this box whose size is determined by the thresholds. The receiver can ignore this contract, but it will be wrong. Now, some people are concerned with doing better than this. For example, look at previous PDUs to try to deduce higher order angular rates, and you can do that to try to refine its position inside this box. But you can't use any

algorithm that will produce a location outside the box; you'll just be wrong.

As a final point of clarification the sender is free to use a low order policy simply by setting the higher order rates to zero. However, as a consequence it will typically send more PDUs. This might be appropriate for a ground vehicle.

Articulated Parts

As specified in the protocol, articulated parts are represented by an azimuth and an elevation. In fact, there are many more cases of articulated parts that cannot be represented in this way. For example, there are many cases of linear motion, submarine masts, antennas, and recoiling guns, for example. Fully general 6 degree freedom motion, the end tip on a refueling drogue, cannot be represented in this manner. The second point is that there is no reference direction defined for azimuth and elevation. This needs to be defined. Further, it's not clearly stated that the motion of articulated parts is a reason to send an entity appearance PDU. This is critical because for articulated parts like, for example, a tank gun, you're quite interested in where it's pointing accurately to within a threshold. You don't want orientation of tank guns to suddenly change when an appearance PDU is sent for another reason. And finally, the thresholds need to be specified for articulated parts and, as I mentioned before, the mechanisms for dead reckoning then may be desirable.

Restricted Data Representation

When we're defining the data fields for DIS we have to be careful, especially for fundamental fields, because we're defining the limits of what can even be represented in the protocol. If we're not sufficiently flexible there are certain domains that just simply cannot be represented, and I think we have a few examples

where the choices have been too restrictive. One is rotation rate. The maximum rotation rate currently supported is 1/2 revolution per second. This is very slow, slower than the roll rate of a high performance aircraft. Another example is world coordinates. Currently the maximum resolution of world coordinates is 1 cm. There are applications, for example, lasers impacting a target, where you may be interested in resolution to greater precision. People may want to do some precision damage calculations and so on, where you'd like a finer resolution than 1 cm. In any case, while it's suitable for most training, I don't think we should be overly restrictive. On the other end of the scale, we do go well out into space, but we don't quite make it to geosynchronous orbit, a relatively significant place. I suggest we be much more flexible and include geosynchronous orbit in the domain of what can be represented.

Third, is the orientation of articulated parts. This is relatively specified down to only about 1.4° or 25 mils, for many applications involving precision pointing of things like gun tubes, radars, lasers, and so forth. You're interested in a much higher precision.

Dynamic Thresholds

The problem here is that basically we need more specification. The idea is a good one, but for example if you see the concept of dynamic thresholds is an Update PDU which allows you to set the thresholds for a target simulator, the concern is if you receive conflicting commands. What do you do, choose the tightest, the loosest, the most recent, or whatever? If you have agents who have different agendas, a network manager or someone interested in precision location, you may introduce oscillations because there are people manipulating the same variable with time delays and pushing it in different directions.

Weapons Fire and Damage Determination

The approach taken in the protocol imposes considerable costs on the receiving simulator. The approach we used in SIMNET was to have two kinds of impact PDUs. One was called an Indirect Fire PDU which dealt with area weapons. It says "A detonation occurred in this area and everyone's responsible for determining if it affected them." There is another class of PDU called an Impact PDU which says, specifically,"I hit you, and here are a bunch of parameters describing that information." In the case of the draft standard there is only a Detonation PDU which is quite similar to the Indirect Fire PDU and it's required to handle both these cases. However, it lacks certain information like the identity of the vehicle that was struck. This information is known to the firing simulator, so we believe we should put it on the network to avoid computation on nearby simulators. In fact, in the case of high performance, high speed vehicles, this can cause a considerable amount of extra computation because when a round impacts on the skin of the vehicle the time it takes the message to get there. In that time the vehicle can have moved a significant distance, which affects damage determination. So first of all, even to determine if you've been hit you have to back project your state, and figure out if the detonation point was on the skin of your vehicle. That's a relatively expensive computation and unnecessary. Further, all vehicles in the vicinity of the detonation will have to do this calculation. Also eliminated are some impact parameters and platform coordinates which also aid in this solution because if you specify the impact in platform coordinates then you automatically know where in your coordinate system you were struck. It also eliminates some parametric information that provides some flexibility on how to do damage with munitions that you don't know about.

Ada Representation

Ada is certainly suitable to define the protocol. However, if you want to treat it as other than pseudocode and actually build a simulator using this Ada code, a couple of issues need to be clarified. One issue is the use of unsigned integers. Unsigned integers are not supported by Ada. However, we already have a working group recommendation from the July meeting that proposes that we get rid of these when they are used for calculations. That will solve the problem. The second issue is that for variant records Ada is a high level language and Ada compilers can arrange the data in memory however they see fit. If you want two Ada simulators to interoperate directly from compiling the same source code, you need to specify the definitions using something called a representation clause, which allows you to specify to the bit level how records are laid out. There are some problems in that the representation clause wasn't in the test suite until December of 88, so all compilers may not be fully compliant. But we certainly want to move in this direction and help people using data based compilers or simulators. The next slide is just an example of a representation clause here. Using the first definition in the protocol.

We have these issues we need to resolve. How are we going to do this in time for December? There are quite a few of them. One approach, of course, is the working groups we have. I believe these issues I've highlighted should be directed to the working groups, and they should be urged to come up with solutions in time. Second, we already have a substantial number of recommendations, some coming out of the July working group meetings and, no doubt, many more will come out of the meetings tomorrow. I think we should have a revised draft of the standard before December so that the working groups can go back over this, and look for interactions (there are several components of the protocol that depend on each other) before the final draft is established. Finally, we should

keep in mind that there are ways in which you can limit the scope of the draft for things that we just don't think we can settle in time. For example, if we make clear the mechanisms for expansion, one of them is simply adding PDUs. We can reassure people that certain needs can eventually be addressed even though we can't address them in time for December. Another issue that should be given high priority in the working groups is that there are some mechanisms that we know we need and we should pursue them at least far enough to know that we have the hooks in the protocol to support them once they are finally resolved.

Finally, if we can't settle on a solution, for some things, like dead reckoning for example, that are just central to the protocol, then I think we should label an interim solution, label it as such and release it in the December protocol. That's all, any questions?

(Question) Just a couple of comments. You were talking about linear motion as far as how the articulated parts record is not able to handle that. We realize that the articulated parts record couldn't cover every single case, but I wonder how important it is to represent the recoiling of guns or the linear motion of an antenna as far as training value is concerned. We can represent everything, but we have to remember what the standard is written for. Although those things can't be represented I don't know that they are really important.

Well, it's really a question for the customers and that's part of, for example, what was pointed out, that we need to survey the requirements of the application. When approach is simply defined what we see is a user application and we need to make sure that we present a protocol that's sufficient to support that.

(Question) Another point was dead reckoning the turret. I agree with that. I wonder what other articulated parts need to be dead

reckoned. Do we need to dead reckon flaps? Do we need to dead reckon doors? Those are things that don't need to be. We probably need to specify which articulated parts should be dead reckoned. The only one that comes to mind is the turret and the gun.

The point is we should include a mechanism so that any articulated part can be dead reckoned. What articulated parts you want to dead reckon depends on the application. In general, you want to dead reckon articulated parts that move frequently and that can be observed. If your application was close formation flying and you look at the guy's flaps to figure out which way he's rolling, I don't know that, but then you'd want to include dead reckoning on those parts.

(Question) As far as the precision of expressing the position of an articulated part, and what you transmit, perhaps you need to know the exact position of a gun to so many mils if you're determining where it's firing. But if you're just trying to represent it as a visual representation you only need a few degrees. You don't need it to be that precise. I think the precision is more for calculating a weapons trajectory rather than representing it visually.

Well, there are other applications. If for example, you want to conduct a plan view and you wanted to know what the guys gunsight picture was when he fired, you need that information or there's no way to reconstruct it, or if you wanted to monitor that in real time. Probably, part of the reason this wasn't addressed is that there's a somewhat artificial distinction between data collection and simulation.

(Question) On the issue of articulated parts, there are certainly numerous cases that folks here can bring up where azimuth and elevation alone are insufficient, and some degree of nearly six degree of freedom capability in articulated parts would be

necessary depending upon what the requirements are. Also, articulated parts can be connected to other articulated parts. You can have a whole series of articulated parts all dangling from each other in the case of refueling drogue from an aircraft. So articulated part's relationship to others also needs to be a part of the PDU.

Agreed.

(Question) I tend to agree with that too, I think that most of these recommendations are things that I can agree with. The only exception perhaps being damage. I think that that's an issue where we have to seriously consider how simulators with differing fidelity are going to interact. If we are all pretty equal, than I agree with you that the emotional advantage of having the person who saw you in his sights when he pulled the trigger compute where on your tank the shell impacted is a very desirable feature. However, I think that for many of the weapons that we're looking at, the effect is going to be too difficult to lay completely on either party. I think there has to be some kind of mechanism established for handing off, especially intelligent weapons which have some kind of feedback system that allows them to be controlled in some terminal homing phase. And in general, I think that the problem with damage is a much more significant problem for interesting weapons, heat rounds, that kind of stuff, than it is for the plain old World War II style bomb. You're right in that the explosion PDU that's described now is really optimal for that World War II kind of bomb. But I think that this is another one of those issues that's more complicated than simply coming out with a few additional parameters. I think this is an area where you need to have some people who are expert on weapons effects. I don't see many up here involved in determining what the weapons effect parameters that people need to know.

I agree that we need a mechanism for target handover as suggested by a recommendation from the July working group. However, I think the suggestion applies as well. If an entity knows where an impact occurs you'd have a way of expressing that, for example, if you handed it off to a designator and the designator knows where the impact is, he should be able to express that to the target.

(Question) This morning someone brought up what I think is one of those line definitions we need to keep in mind here. When are we stepping over what should be proficiency training or tactical training, if you will, on a broader scope, versus skills training? I share Chris' concerns about the recoiling rifle and gun turret positions and things. I think that's a foolish extravagance and it's going to slow the process down on how to mechanize it. I go back to one point I made earlier, before lunch. We seem to have an open checking account on all these bits we're going to broadcast over the network, I'm concerned from the air-to-air war standpoint of rapidly changing parameters and things that the throughput time is going to be gross. But that notwithstanding, I don't need to fly from Phoenix, Arizona in my visual simulator and meet somebody out over Oklahoma and worry where in the hell his flaps are or whether his gear is down, and those kinds of things. Again, I think that's far beyond the scope, which as ASD has pointed out has not been yet defined. That's not the point of the tactical C3 training, all these things of whether I shoot him with his flaps down. If I shoot him with his gear down, I probably shot the Accords of 47 in Geneva. But, the point is, in listening and not being an army expert, I do think from a close air support role in the A-10 business, if I were hurling my body at the ground it would be of some interest to me to know that generally there is a turret in my direction, but only that I see a muzzle flash that he's probably shooting at me. And we go back to another point made this morning. Cther than the skills versus proficiency, if I really want that kind of training and need to know that M1 is pointing

right at me then you're probably talking about another entity within my own simulation system that is a high fidelity system. It matters if he's pointing at my compatriot over this hill and I can pursue him aggressively as opposed to him pointing right at me. Again, if I'm traveling 350 miles to the battlefront it's not important to somebody in the air. If we're going to reproduce the Monitor and the Merrimack, yes, then I need to know that gun #3 on turret #2 of a ship is at a lower elevation. But if I'm just talking war gaming, tactical training, C3 training, and those kinds of things, the fact that the ship muzzle flashed and shot in my direction is all that's significant. Simplifying what I'm saying, the articulation down to azimuth in relation to platform coordinates the elevation of a turret is, as one or two others have mentioned, all I see that's necessary, in order to get through the bottlenecks and get this thing with some reasonable speed.

As you indicated, there is sort of a tradeoff between the number of applications we fulfill and cost of the applications as a consequence of doing that. Again, as a recommendation we should be quite explicit about the scope of systems that we're trying to support, Then at least people can argue specifically about what that scope should be.

"Floating Point is Faster and More Flexible Than Fixed Point," Joshua Smith, BBN

I'm Joshua Smith from BBN. There's the proposal. The issue at hand is whether it is really a good idea to use fixed point instead of floating point for representing all the various fractional numbers that we are going to see in the protocol. My proposal is that we should, in fact, look at using floating point, the IEEE standard, binary, floating point representation wherever it's natural in the protocol. What I'd like to discuss is the extensibility of using floating point. In some cases where floating is actually faster than fixed point. I'd like to discuss the results when I did some tests to find out whether IEEE was really a good format for representing floating point, as there are several. And then finally I'll give some specific recommendations as to where I think we could look at using floating point numbers.

To give you a little bit of grounding in the concern, thirty-two bit fixed point which is used for location and other bit-linked fixed points are used for other things, like the rotation of an articulated part or other things like that. In general, the way that that is interpreted is, in the protocol data unit you specify, the numerator of a fraction and the denominator of that fraction is specified a priori. In the case of location, for example, the denominator is 100 where the numerator is the number of meters you are in geocentric coordinates. To give you an idea of the range of those values (well I think Richard already has just previously) you can go almost out to geosynchronous orbit and you have a resolution of 1 centimeter. That is nine significant digits. An alternate format is the 32-bit floating point. It takes up the same number of bytes on the network but as you can see the range is much more extreme. You can go way out and you can go down to a very high resolution, but obviously you are loosing something for that extra bit of information. You are losing a couple of significant digits. That's seven significant digits based on 32-bit floating point.

And I show you the layout of the IEEE floating point. The 64-bit floating point is more of the same. It's just that you are representing the thing in 8-bytes and it gives you larger range probably more range, than we would ever need, but it does give you a higher resolution. Obviously still at issue is what the appropriate resolution this protocol needs to have.

What I did is I went to look at the two critical issues that I could identify which would have led the folks down here to choose fixed point over floating point. The two issues I was able to identify were, one, the belief that calculations and fixed point are going to be faster than those in floating point. The second issue is that the representation of floating point is not in any way standard. Therefore, it may be better to choose something simpler that everyone understands, like fixed point. What I started with was a good old Motorola 68000 family computer. I tried a couple of them and what you have here are two different trials on each piece of machine. I urge you not to compare machines from these numbers because that is not a fair way to compare machines. But this is merely to test how long it take you to do register to register operations; to do an addition of two integers; to dc an addition of two double-precision floating point numbers and so on. The numbers that you see here for addition and subtraction are the ones I think everyone would be expecting. That is, that double-precision takes a lot longer because with double-precision you have 8 bytes. You have a lot of special issues to concern yourself with and it just takes longer to do. In the case of multiplication and division it's starting to get a little fuzzy. Now this has a floating point co-processor running and the integer operations are still faster, but they are not a whole lot faster than the double-precision operations.

I1 you then look at some of the newer architectures (the two that I have on the slide are the spark from Sun MicroSystems and MIPS architecture), what you start to see is that the old standby

that integers are faster just isn't true anymore. In fact, you can see that with addition and subtraction you get a little speed advantage by using integer. With multiplication and division you are much better off using the higher precision, double precision floating point. It's counter-intuitive, but it's driven by a lot of specific attributes of new architecture. Things like the old overhead of trading off the operation to another processor that we used to have with co-processing systems is starting to disappear. The amount of silicon that you have available to dedicate for a particular task is starting to become very critical. And since high-precision, floating point operations are important to a lot of customers, these folks are focusing a lot more of their attention on how fast they can get the double-precision stuff to go.

So you see with the MIPS, even integer and double multiplication, you've got the same time. With division you are doing better time with the double-precision. So I don't want to exclude any particular architecture. I want people to understand that not all architectures share the same advantages when it comes to choosing one format over another. A lot of the newer architectures you are starting to see are, in fact, giving preferential treatment to a floating point format.

The other issue which I was concerned with is whether IEEE is a good format. And what I did (you can read the specifics in the paper) was dream up the worst possible format, the most different format from my IEEE I could. This is for somebody who wants to use floating point internally, but their format is really, really different. And I then worked out a routine to convert from IEEE to that format using a bunch of shifts, masks, adds, and so on. And then I took the amount of time it would take (and this is on a MIPS) to take a fixed point number and translate that to an internal floating point. You can see that it was in fact, faster to do a bunch of shifts, adds, and so on, to translate the floating point format than it is to start from a fixed point and just do a

simple multiply to get it into floating point format. The significance of this is that a lot of simulations need to use floating point internally for things like transcendental functions or various things that you are going to run into. You are looking at a need for not fixed point internally, but a need for floating point. In that case it turns out that you are better off starting with floating point to begin with. And if your bits are a little different, and you are using two's-compliment here and sine-magnitude there, it really doesn't matter. The big issue is that you are starting with a floating point representation; you can manage to stay within that and doing that conversion can actually be rather cheap.

So, to jump right into my recommendations; I went through the protocol and identified the areas where fixed point was being used and where I think floating point should be considered. In fact, I would strongly urge that floating point be used. With the exception of angles, all of these fields have been proven to be feasible using floating point in SIMNET. For angles we have used what people call, "BAMs". You break the circle into n-different pieces, and you say which piece of the circle you are talking about, for example "my turret is pointing into the two hundred and fifty-six thousandth quadrant, or whatever of the circle". That has turned out to be kind of a problem on a machine like we are starting to deal with where the obvious motivator for that is that you can then take a lookup table and find all of your transcendental functions on those angles. On a lot of hardware, you are better off just using the internal sine, cosine, tangent functions than you are doing lookup tables; that on a RISC machine memory is what kills you. You are much better off just using a few extra cycles to actually compute the good value. Therefore, I would suggest that we look at using radians for angles, angular velocity in radians per second, various articulated parts. If they need or have an angle you use radians or if they need to have an extension, you use meters. I wasn't even going to try and guess

what the right units for electromagnetic, characteristics are, but I am sure that there are some fairly well accepted ones. Most of them would not be difficult to find good ranges of fixed point numbers that you use for them (things like power). Vehicle rotation representation would depend on what vehicle rotation scheme you use. If you are going to use Euler angles, go with radians; otherwise there are no dimensions involved. Your fractions, I think, would probably be best served with floating point numbers, linear acceleration in good old meters per second, per second. Linear velocity in meters per second. Supply quantities representation depends on what exactly you're supplying. It may be easier to measure fuel, for example, with a floating point representation than to try and guess the proper amounts of fuel that can be transferred between entities. So, with things like that you may want to be a little flexible and allow the use of whatever formula is most appropriate in that case. For world coordinates, that is, the location of a particular simulated entity, I strongly support the use of floating point and at SIMNET we've used double-precision floating point. We've had no problems. In fact, we have a machine which runs as a part of just about every simulator in SIMNET, that has no floating point hardware whatsoever, that is doing a position-based filter based on these double-precision floating point numbers. In fact, it keeps up just fine. It handled thousand packet-per-second packet rates during a recent large-scale exercise with no additional floating point hardware whatsoever. So since the extra precision of double-precision could conceivably become very useful in the near-term future, I think restricting yourself to single-precision in that case would probably be a bad idea. Questions?

(Question) Let me pose a couple of things that you don't touch on very much. It's too bad I gave my slide away. I did an actual, honest-to-goodness experiment where I looked at formulas that were interesting, the Euler angles to Quaternians and back and forth formulas, as opposed to simply looking at the instructions. On

that example, for those of you who remember that slide from this morning, it is still just about three times faster to do that particular manipulation with integers than it is to use it with floating points. And that is in spite of the fact that the floating point numbers take advantage of my 688082s floating point co-processor's embedded trigonometric functions, and I have to use software to do sines and cosines with my fixed point numbers. The difference is the particular instructions that you looked at, (I don't know, maybe they come from the way that your machine's pipeline instructions in doing ten thousand adds in a row gets pipelined differently than doing ten thousand floating points instructions in a row), but we use INTEL 860s, which is another RISC-like kind of processor and the results that we show there are very similar in that they're about 2-3 times faster to do integer instructions than do floating point instructions. So although that factor is certainly going to become less important in the future I don't think that it has gone away yet. I also don't think that all the people who currently are interested in having their simulators work in this simulation internet are in an economic position to afford to go out and get a new RISC computer-based simulator. Historically, the numbers have been much, much stronger in favor of integer versus floating point. That, to me, is really not the issue. It's kind of like the discussion that we had earlier today about Quaternians versus Euler angles. If you want to use floating point internal to your simulator I think that's wonderful. I certainly intend to. But I still think that you should transmit integers over the network because I think from a standardization point of view, you want to standardize on the most directly meaningful notation that can be converted with reasonable expense into forms that people can use. It is still considerably more expensive to make those conversions, as you showed quite well up there than it is to do the operations in a lot of cases.

You talk about going to double-precision floating point numbers where you're going to expend twice as many bits worth of

bandwidth on parameters that are transmitted very, very frequently in Vehicle Appearance PDUs, in particular, for a 50% improvement in the number of significant digits and some improvement in dynamic range. I think that I'm not willing to have all of my world coordinate numbers go up 100% because we have this agreement in the working group as to whether it should be 32nds of a meter, which is what I think that we recommended, or centimeters. To me that's the kind of PDU that, since it's going to represent 95% of the traffic on my network, I want to expend the extra engineering calories to make sure that I store in as compact and efficient form as I can. So I agree that the benchmarks that you have are probably accurate results. I am not convinced that this is the argument that pushes me away from well-understood, well-known fixed point numbers into floating point numbers. And I certainly don't think that there is sufficient performance improvement to be gained that I am willing to make a reasonable chunk of my numbers twice as big just to avoid having to spend a few more hours arguing in some minuscule working group session on representations even though many hours have gone past. So I guess I'm still unconvinced. That should be my question, to put one at the end. The official question, I guess, is what do you think of that?

(Question) Well, let me try and go through the points as I can remember them. Please remind me if I missed any. The issue of whether integer is faster or slower than floating point is only a real issue if you can use the numbers exactly as they are coming into your machine. If you are going to have to do any sort of translation of those numbers when they come in, then their format on the network is really immaterial as far as he is concerned. You can translate them into your favorite internal format however you like. Suppose that you have a fixed point number representing, for example centimeters and you need meters internally. You're going to have to do an integer multiply, typically a very expensive operation on some machines. On some other machines it's very cheap. In contrast, maybe you have a floating point number coming

in, in meters, and you want something. What you are going to have to do is shift around the bits, and you're going to have to come up with an answer. I don't think that the specific speed should be a terribly great concern. The assumption that a low cost, low fidelity simulator is going to be able to act better using integer operations is unproven, and my guess would probably not even be true. That in fact, the floating point numbers is probably a better solution for a lot of low cost simulators.

(Question) Well, maybe we should just do them one at a time. I agree. You're right. It's certainly an improvement, and it's something that should be evaluated experimentally. But I think that when you consider the enormous amount of traffic that exists on a network, and the fact that what you do with most of it is two greater-thans to see if it's close enough to be within your field of regard, those greater-thans in world coordinate systems are going to work much more quickly with integers than with floating point numbers and you're right, I perhaps could be swayed by some really impressive empirical evidence to the contrary but...

Okay. What you're saying is it's in fact more than just two greater-than comparisons. You're trying to find out whether it's within a certain range, for example, if you are going to do a filtering. I think that's what you are referring to specifically. Well, to find whether it's between a range, you're looking at least two coordinates, probably all three coordinates since we are going with geocentric coordinates. The comparisons involved will have to be done whether or not it's an integer or floating point. I've shown that it really doesn't take a lot to get the floating point down into a format that you can compare even if you don't have any floating point hardware. And we've proved in SIMNET that in fact you can handle very large data rates with very slow processors, you know, a 68010 running on some little ETHERNET device. It can handle this conversion just fine. So,I think to throw away the flexibility of floating point just because you are concerned about

the extra overhead of three or four more shifts, adds, and that sort of thing on two particular processors probably is not a wise decision.

The other issue you brought up was the question of whether you should use 8-bytes to represent your locations. If you don't think that 8-bytes, and if the group as a concensus doesn't think that 8-bytes is really necessary, and the saving of 4-bytes on the network, (for each coordinate, obviously), is going to lead to some great performance improvements in the network, then by all means use 32-bit floating point. I don't really think that is a significantly large issue.

(Question) What about the fact that floating point only gives you seven digits worth of different answers between the center of the earth.

But what you're looking at you're comparing to the alternative which I think we all accept as probably not adequate either. That is, if you break down the system into 32-bit fixed point you're going to come up with nine (9) significant digits. Why is nine so much better than seven? I think neither of them are very good. I think you have to have more accuracy. I think it's an important point. In fact, there was another position paper presented last time around where somebody suggested that 40-bit fixed point would be a better solution. Well, your only going to be able to do just a couple of comparisons with 40-bits. If 40-bits are really needed you've got a lot of extra operations involved.

(Question) I have a real quick comment to make. My name is Steve Swaine. We have a tactical situation on the large-scale simulations that we have been running, in a tactical environment for quite some time. And when the draft standard came out I showed it to our engineers that work in the tactical environment. One of the major complaints they had was that a long time ago they decided

that they needed the resolution of full 8-byte floating point in order to get the job done. And on the basis of that alone they said we will not be able to use this standard for any internal use because we need the resolution. They've discovered that they do need the resolution.

Yes, we've come to that conclusion in SIMNET that 4-bytes just wasn't enough. And to get two more significant digits out with fixed point probably isn't going to solve your problem.

"Query Protocol For Distributed Interactive Simulation," R. Saunders & M. Robkins, Hughes Simulation Systems

I would like to talk today about the advantages of adding some query-based protocol data units to the current draft standard. I would like to talk a little bit about some of the topological assumptions that went into the draft standard. Maybe we should stop for just a point here to say that I do agree that these are things that do need to be written down and understood. I think that these are assumptions that have been argued out in the last couple of these meetings. People who don't have the benefit of being at these meetings are going to find these assumptions more difficult to catch up with, in particular, some of the possible alternative topologies that I think people are going to propose, and a lot of the implementations of this distributed simulation. Then I want to talk a little bit about some of the alternatives for how queries could fit into that, and just touch for a second on some of the effects of analyzing traffic in that kind of a situation.

The real classic assumption that I have gone into with the development of the draft standard is that the simulator interface, the thing that's actually described in these PDUs, is some kind of ISO layer seven application protocol. And the network itself is some kind of a continuous, multi-cast medium where all the bits and all the packets go to all of the players, perhaps through some kind of routers to change them into different kinds of electrical configurations, but essentially everybody is going to see everything. The advantage of this set of assumptions is that this is what SIMNET does. And there's also a real common, really reasonably priced implementation of this, and that's ETHERNET. The con that I have with this, and this is the most significant con, is that this just isn't the way most military organizations work. They just don't have the concept that everyone is an equal peer and that everyone's information is equally interesting. Hierarchies

are very stiff in the Military and they are very important to the way that those units function. They also have a con in that each simulator must process each packet, at least to some trivial degree to find out that it's not interesting to them. This puts you in a situation where the bandwidth required for your medium to transmit these packets has unbounded scope when you think that there could be an unlimited number of semi-automated good guys and bad guys ultimately controlled at a very high level by some kind of JESS-based simulation.

What I think you are going to see and, in fact, you are already seeing if you look at some of the WAREX Viewgraphs where BB&N has talked about how that exercise is actually done, is a hierarchical kind of structure where there is going to be some kind of intelligence used in consolidating traffic. And there's going to be some kind of distinguished entities that have special jobs and special roles in the simulation, and those people are going to be treated a little bit differently. What I think is a real advantage of this, if we choose to take it this way, is that you can make something that directly follows the military organization that you are trying to work with in this exercise. This gives you some of the benefit of the Military's many years of deciding how many platoons ought to be in a company and how many company's...and so on and so forth. So you get benefits from the fact that they have looked at those kind of problems and have empirically determined that this is the kind of force-structure that gives the most reasonable compromise between broad scope of control and a workload that you can expect reasonable entities to evaluate. The con is you have to add some of these pieces, and you have to put some intelligence into them. But practically, you probably have to put some intelligence into them anyway so they can act as an application gateway, if that's your preferred term, to filter out information that just doesn't need to be transmitted across the network. There are already plenty of distinguished entities in SIMNET; there's the Data Logger and there's the Stealth Vehicle and

other kind of systems like that which are handled differently by people. So I'm not really talking about a significant change in the way we think about things, but I think these kind of topologies are what are going to be used in real systems and we need the ability to do things in the standard that make this kind of a strategy as beneficial as it can be.

One of the things I think we can look at as an example to isolate queries from the current draft standard protocol is that the draft standard assumes that periodically you re-transmit all of the redundant data so that you get all of the update that you could ever use. This is kind of neat in that it means that passive listening for some small amount of time, like five seconds, gives you the details on the entire simulated world, everything that there is out there. That's used in SIMNET for people joining exercises and it's used in some other situations, but we decide what the real benefit is of being able to have that kind of feature because it is enormously expensive in data bandwidth. It wastes a lot of the benefits of some of those alternate topologies because now you have a tank that stopped, whose crew is in some kind of briefing, that is still emitting packets all of the time because it wants its friends to continue to draw it in case you drive up near it. It doesn't want to be deleted from the exercise. It continues to transmit all the time where it is even though the motor's off, there aren't any people in it and its not going to move. What we propose is that you have some kind of query PDUs that you add to cut down on the redundant transmissions. That has the big advantage of minimizing the data bandwidth and not sending us unchanged data all the time. It has a side benefit that you can use a different topology to significantly improve the performance of your network. You can put a little bit of memory into those company computers. You can put a little bit of memory into your application gateways and not send the query all the way back to the other side of the planet when you can have that kind of information remembered in just one place; although the example that I use all

of the time is querying someone about their location because I don't want to tread on the toes of the database people. I suspect that the database people are going to want to use the network to transmit dynamic terrain changes and information like that, and to simply expect everyone to store all of that information is unreasonable. It's going to have to be centralized in some kind of a distinguished node and queried from them at some point in time as an efficiency point of view. The deficiency of this approach is that you have to be a little bit active in your listening if you want to understand a simulated world that was started before you woke up. I think that the level of the time that you are going to have new people joining your exercise is sufficiently low that requiring a little bit of thinking on their part is not an unreasonable prerequisite to joining the exercise.

In general, queries can have a couple of different characteristics. They can be specific to someone or they can be general, in that you can specify that I want to know where you are, or I can specify that I want to know something based on other factors, such as everybody within a certain radius of some location. You shift some of the work onto some of the outside people to keep track of who they are, and that kind of stuff. They can request some kind of specific data that you already sent, like vehicle appearance data, so that I can have some kind of a one-shot update outside of the dead reckoning bounds so that I don't have to send you some kind of an "update your frequency of PDUs" messages to get you to transmit one and send it back. They can also request very specific data. I think the important thing about specific data is a flexibility one. We can't really tell ahead of time what all the parameters are that we're going to want to address. I think that's a point that has been argued by a third of the people who have been up here. But by letting you put that kind of information into a query, you allow the capability that some distinguished person that you add to your network later, could provide that information to simulators without going out and

modifying. The example that we use all the time is the acoustic signature of a helicopter. You'd hate to have to lump that on to all the helicopters that you ever buy just in case they fly over water where your Navy sonar transceivers can hear them. What you'd want is if the Navy wants to participate in an exercise and they want to use those sonar transceivers, they have to identify some kind of distinguished part of the simulation network, hopefully connected locally to the sonar simulators themselves, that is going to provide that kind of acoustic parameter information on any old helicopter that flies by and let you leave that completely within the scope of the people who have been working on the sonar-affected side of the system, and not lump all that work off on the other people who just happen to be in the helicopter business. I also think that there are some things that we are going to want to ask about in general. I've just mentioned weather and dynamic terrain as things which I think are going to have to be computed, because the idea of storing all of the possible combinations ahead of time in a database is going to prove untenable, although that is yet to be the recommendation of the database group.

So here's an example of how a query could work. A query could work very simply in a real dumb kind of network where everybody was seeing everything anyway. When you transmit it's transmitted uniformly to everybody, and the person that you are talking to responds back. Or, you could have the capability of a more intelligent kind of system querying and the first person along the line that knows the answer sends it back to you. This is not a requirement. I wouldn't want this put in the standard as something you had to do. But I'd want to leave this open to the people who were making this network and selling it to you who would be able to put this kind of information in without violating any of the requirements that are in the standard. I suspect that when we start to talking about how PDUs have to be distributed and we start putting that kind of information in the standard, doing something like this to short circuit the system, by saying that you are going

to ignore all the Vehicle Appearance PDUs that come out of the sky, is going to be considered non-standard compliant, and when you test your machine it is going to fail. That is something I think we desperately need to avoid.

One of the things that is very important, I think, in choosing which things you want to remember, which things you want to forward, which things you want to query, is the real analysis of the traffic patterns of the particular problem that you're working. I think that this kind of a rough analysis can be used at best to support the idea of keeping options open. But I don't think it can be used to convince anybody. It certainly couldn't be used to convince me. What you are going to have to have is real measurements of the particular problem that you are solving, to tell yourself whether this is the right kind of information to store in local area nodes, this is the right information to store in some company-level computer. I believe that there is no point in trying to do that analysis now because I don't believe that one side is going to address everybody's problems. I don't think there is going to be any reasonable way to get complete agreement on what the correct level of things to query for, and what the correct level of things to transmit all of the time in Vehicle Appearance packets is. That's why I think that you need to add the concept of a query-based system that is soft, that allows the querier to specify what information he needs so that you give the flexibility to people who are implementing those kind of networks in the future. Trying to tie together simulators built for this MIL standard, built by someone else that they don't want to have to change into a specific training problem, some kind of control is to work that out.

I would recommend that you add some PDUs. I talk about six in the paper. A lot of people have asked me about an oblique remark I make in the paper. I'll show you an example of what I was talking about there in a second. I think you need to add some

kinds of PDUs that let you query for information that you don't know. The strategy for using them has to be that you set some default value into your simulator, send out a query and if you get an answer back that gives you some more specific data, you use that. And if you don't, then you just use your default. That makes them completely transparent. People who don't understand them and don't have any distinguished entities to fill their data in are not dead but your simulation is just going to operate with some standard defaults instead of something that is tailored to this specific problem. I think we need to change our thinking on what's a reasonable idle re-transmission time of something more like five minutes. I'm not completely against the concept of keep-alive kinds of messages, but I think that they should be an unmeasurably small fraction of the network traffic. We need to really look at how effective query is going to be for some of the kinds of problems we haven't gotten around to working on yet, for example, initialization of the system and finding the initial state of a game that you are trying to join.

I talked about six PDUs in the paper, and I didn't talk about formats of them because, frankly, I don't want to get into an argument about formats. I think that the people of IST have done a wonderful job of picking formats. But I said a little thing in the paper "or you could make it all into one PDU", and people have told me it was a little unclear. So here's an example of how you can make that into one PDU. I'm not recommending this. This is just an example of what I was talking about. I still think that the people at IST should decide on what the format is after doing some empirical work. You could say that you want to know the answer to a question, and you have some little kind of control word that's bit-encoded because we can tell ahead of time all of the different categories that we are going to specify in the standard for this PDU. And depending on which of these bits you have on, you have some or all of these characteristics and you are supposed to respond with either an Appearance PDU or a Definition PDU based

upon whether you pass all of the characteristics specified in here with these little bits on. You could develop something like that if you are concerned that six PDUs is a waste of PDUs, an unnecessary complication of things. You might be right. I really don't want to get into arguing about what the correct way to PDU-apply that because I think that is a problem that is easily enough worked by the people at IST. Questions?

(Question) In the current SIMNET implementation one of the things that the reissue is used for is if it doesn't occur they use that to take the entities off the network where somebody may have tripped over the power cord, not where somebody might have gone down gracefully getting shot. But if somebody trips over the power cord you may argue then in a small simulation that that kind of thing isn't necessary, because if somebody goes down then that invalidates your task, and you need to run it again anyway. But in a large-scale simulation where you're talking hundreds of tanks and things together all over the country, I don't think you want to bring down your entire network if one person goes down, and I don't think you want to have one tank being dead-reckoned off into the ocean either. Do you have any suggestions for how, in an environment where you don't do idle retransmissions so often, you would handle that?

Well, again, that is why I've tried to stay away from saying that we should get rid of re-transmissions altogether. I think that at five minutes you're much less likely to be interfered with by somebody who kicks his power cord out than if you never do it. But I think that five minutes is probably plenty of time for you to wander around. You really don't object to that. But again, you know, I'm thinking five minutes completely at random. This is the kind of thing that we need to look at some reasonable example cases and find out what the real parameters for that are. Maybe again, that's some kind of thing that needs to be said in the Threshold PDU so that along with setting some kind of minimum values you can

set some kind of maximum values. Because maybe that needs to be adjustable on an exercise basis but to me those are the kind of things you have to look at empirically and find out. You really can't guess.

(Question) It appears to me that you're trying to reduce bandwidth and I think that seems, on one end, to be a very noble idea. You say, however, that this query protocol allows easy entry into running scenarios, I envision two situations where bandwidth would be incredibly large. Take the first example where there are five simulators on the network, and all of a sudden some semi-automated force unit comes up, creates 200 vehicles immediately, and all of a sudden 200 vehicle appearance packets are sent out to the net, and you have a spike. Now that happens all the time. Consider however, the opposite scenario where you already have two or three hundred vehicles on the network and an operator decides he wants to start up five more vehicles. He says, "alright, start up that one, that one. Five vehicles start off and each one of those five vehicles as they start up issues a query-all PDU to every simulator on the network to get their appearance and their characteristics. And that entire sub-net, as it were, is flooded with PDUs. Whereas, if they had just waited five seconds they would have received this information.

Well, that's clearly a case where you need to do something. That's exactly, though, the kind of example that I am thinking of. Pretend there are 200 hundred guys; I just didn't draw them all. The first time you turn on this little guy he queries his little company computer. His little company computer that just got turned on now too doesn't know and has to go around as far as this guy, and there is, for a period of time, two hundred tanks worth of traffic to here. But when you turn the second guy on all the traffic that you're consuming there is a little bit of traffic on this local ETHERNET which you probably sized correctly to support some ten or twelve tanks that can afford that. I'm not saying you

have to do that. But I'm saying that that's clearly a kind of strategy that you want to be able to support in the standard. You're right. If you want to be completely passive maybe you have to wait five minutes to hear all of them, and you should tell people that the strategy for joining a session in progress is that you are going to wait for five minutes. Maybe the strategy for joining a session in progress is that you are going to transmit that you want to know all the people within two thousand meters of you, PDU. That strategy has to make it into the standard or the rationale somewhere because otherwise it's one of those things that each of us is going to do differently, and the people who are in charge of first using our heterogeneous simulators are going to take some penalties. You are absolutely right about that. If this is done incorrectly it is going to make things worse. The other thing that you have to consider is how often do you really turn on a whole new troop of tanks? Maybe making them wait for five minutes isn't a significant penalty.

(Question) No, but often times the situation occurs that for one tank out of a group of a thousand someone will trip over a power cord. If there are a thousand vehicles on the network or even a hundred on his local area network, he does go down for a few seconds and comes back up. And in a long exercise it's maybe 8-12 hours long which is not an unreasonable one. That can happen numerous times.

Sure. Those are cases where for a little bit of time his little local ETHERNET is going to have a little bit of excess traffic while people stuff his brains back into him, but better that than the whole network going away.

(Question) I was just going to observe here that by pointing to the company filters and saying, "well, it's done here", you've really just moved the problem into another box, but not made the problem go away. Nor have you eliminated the need for the protocol

to somehow provide provisions for supporting these queries for insuring, for example, that if information does change then things that cache that information, like company filters, are notified of the changes.

Oh, yes. I agree. This is not an intention to make any kind of problem go away. I think the real objective here is to try and allow people a little bit more generality to hierarchicalize their solution to that problem on a case by case basis that varies with the design of their simulator, instead of forcing everyone to go into the only solution that's supported now. Now you have to buy a network with enough bandwidth that you can hear all the targets, which with FDDI is technically possible. It's just expensive. You have to then use the strategy that you'll just listen for five seconds. There are pros and cons to that. I am just trying to buy us a way where we can have a little bit more case by case flexibility in this.

Importance of Experimental Evaluation in Protocol Design, R. Rabines & A. Pope, BBN

In this presentation, I will advocate the position that experimental evaluation should become an integral part of the protocol development process. I will first give you an overview of our position. Then I will highlight some of the benefits of experimental evaluation. Then I will give a brief description of the way experimental evaluation has been an integral part of the way BB&N has developed SIMNET protocols. Finally, I will summarize recommendations.

We believe there is a real potential to develop a set of standards that might not be able to effectively support the kind of applications that they were intended to. This could lead to degrees in credibility of a standard development process and will most likely delay the incorporation of this standard into real applications. We believe that this risk is primarily caused by the fact that protocol design is a very complicated activity. It involves and requires very complex engineering tradeoffs that cannot be performed adequately on the basis of pencil and paper analysis. The presentation that preceded me clearly showed that. There is a lot of very conflicting, very complex issues that cannot be evaluated just on a purely theoretical basis. A solution that has worked for us very well in the development of the SIMNET protocols has been experimental evaluation. We believe this should be incorporated into the development of the protocol standard and that critical aspects of the protocol should be first evaluated in a test bed before being standardized. Some of those have been pointed out by Richard Schaffer.

First, in order to do experimental evaluation, we need to provide some rapid product implementations of certain protocol features. Modern design methodology has encouraged the use of early implementation in order to better define requirements and,

more importantly, better manage the risk of developing a new product. We suggest that multiple organizations should begin to rapidly prototype these standards or at least certain critical aspects of them depending on resources available. We believe multiple implementations should be encouraged because it would help to clarify various aspects of the protocol. We do acknowledge that there are resource limitations and we cannot in any practical fashion prototype the whole draft standard before acceptance of the standard. However, there are certain critical areas that do warrant further evaluation through experimental techniques.

Furthermore, we believe that this prototype of limitations should be evaluated in the context of an applications test bed. This test bed should characterize the environment in which this new protocol feature is going to be used. The test bed would help designers identify critical areas, and will give them an indication of how well this new feature works with the existing features or proposed features. In addition, it will allow, given that you have the resources, evaluation of alternate designs against a common test bed.

In the development of SIMNET protocols we have used experimental evaluation as an integral part of it. We usually analyze the potential impact of a new protocol feature based on pencil and paper analysis. When designers are very comfortable that they do understand, that they do have some insights into how this will affect the distributed simulation environment, they will judge certain pieces to be candidates for rapid prototype evaluation. These pieces of partial implementations will be evaluated in a laboratory test bed. We will do this in a rapid prototyping loop. We will design a little, code a little, then evaluate. Every time we go through this loop our design trade-offs will become more meaningful. This test bed has the unique characteristic that it offers the possibility of playback of exercises, actual simulation exercises from previous SIMNET proof

of concepts, and actual exercises, both training and developmental. We're convinced that this has served us well to minimize the risk of spending time and resources in fully developing solutions that might have turned out to be unworkable in any sort of real sense. It is worth emphasizing that we have used the technique not as much to find an optimal solution or a best solution as much as we have used it to find a solution that would work, that was good enough, and that would allow us to support the requirement, meet the requirement to in a practical fashion.

One particular area in which this technique has proven invaluable has been dead reckoning algorithms. Just based on discussion we've had you know that is one of the more complicated aspects of protocol design because it involves some of the more valuable or critical factors of implementation, like the network traffic and the amount of computation you want to have every single simulator device do, and the precision with which you want this measure to be carried out. Furthermore, the dead reckoning approximation algorithm depends on the vehicle class. What is right for a tank might not be right for an aircraft. It involves very subtle and complex interactions among several of these factors: computation, network traffic, and position. One particular case in which it was not obvious to us what the result would be was application of higher order derivatives of vehicle motion for approximating the behavior of tanks. We figured there would be substantial reduction in traffic. As is obvious, there was not a very substantial reduction. The behavior of the tank is very affected by the underlying terrain. However, in the aircraft situation we have discovered that the intuitive feeling that high ordered derivatives would significantly reduce the network traffic has proven to be correct.

Finally, I'd like to summarize our recommendations. We believe that the best way to minimize the risk of standardizing on a set of protocol standards that may not be effective in supporting

the kind of applications that they are intended to is to evaluate new and complex protocol features through rapid prototype of limitations, and to measure various alternatives. One we've been discussing at length through several of these presentations is an applications test bed that characterizes the intended environment. Any questions?

(Question) I just wanted to make a suggestion. I agree wholeheartedly with what you are talking about. IST is taking, some very modest steps to do prototyping and experimental design and validation. I would invite any other organization, or any other government agency (I know the Air Force is interested in that) to get a group together aside from this meeting so that we maximize the utility of our meetings. If we are going to duplicate things we do that consciously and not unintentionally. So, I wholeheartedly support this idea.

(Question) I am Amnon Katz and what I'd like to say is there is an inherent danger in test beds that are specifically designed for this purpose because you may be imposing the properties of the test bed on your experiments and on the results. So I think it's very important to have (I mean I totally agree that it has to be tested and it could prove totally unworkable if not tested), but I think that the testing must be on a collection of existing and independently designed simulators. I'd like to mention that at McDonnell/Douglas Helicopter Company we have been doing this, networking our high-fidelity helicopter simulators with others by means where we don't design the simulator for the networking, but rather we hook the network into an existing simulator. We have done this very successfully with Bell Helicopter and we, frankly, are looking for other partners to do this kind of experiment. We are proposing to, in fact, carry out the draft as soon as it crystallizes to the right extent. We are talking to IST about doing this, but I would encourage any other prospective partners who would like to do it with us to contact me.

(Question) Jim Hammond. I also support that for the Navy. I think the Navy community would also like to take our existing simulators and try your protocols out at possibly our BFIT Lab at NOSC or other places in working with you rather than depend on one particular entity at your level to do all the protocols. We'd like to do some independent tests on them when you get it in a little better shape next year, maybe.

All testing is welcome, I am sure. Thank you.

BFIT Presentation, LCDR Dennis McBride, DARPA

I am going to be merciful here at the end of the day and only speak for a couple of minutes. In fact, I'm not going to give a presentation. The main intent is to give you an opportunity if you want to stay behind to look at the draft final tape that summarizes the BFIT exercise which was co-sponsored by DARPA and the Navy. But before I do that I want to go back and seize an opportunity. Col. Mark Pullin has joined us. There were a couple of questions that came up this morning that I thought should be answered by him, and now that he's here I thought we ought to do that very quickly.

To stimulate your memory, the issue was the business of war games. And we won't go into a long brief about how we're going to internet war games with human simulations, and simulators but at this point during Bruce McDonald's briefing, the question came up about the standardization of war games and war game entry into the battlefield as we were talking about the standardization process. That's just to bring your collective consciousness back to that point in time. With that, I'll turn it over to Mark to discuss. I think he's got a total of one slide there that may clarify for you or to answer your questions about it. Then I'm going to show the BFIT tape, or actually Tom Tiernan from NOSC has a better version of the tape which will be shown.

(Mark Pullin) Okay, thank you Dennis. I think the main reason that Dennis wanted me to speak was to prove that I exist. Because there are undoubtedly those in this audience that don't believe I talk. I'm the mythical Mark Pullin. I have been involved for several years now at DARPA in Advanced Parallel Computing and Advanced Networking and Advanced Applications of Information Technology. Now I have a new challenge. I'm in the Tactical Technology Office as the Deputy Director, and looking forward very much to working with this new challenge. I'd like to underscore what the last presentation said by pointing out some experience

that we've had in the networking area where the issues of protocols obviously pre-date simulation protocols by a good bit. That is, I think you are on exactly the right track here. When the INTERNET protocols were put together different groups in the DARPA networking research community did just the sorts of things that were just brought up. They got together, drafted the protocols, took them through phases of concreteness, and at about the one-year out point, everyone was busy implementing and trying to interwork and found out a tremendous amount you just wouldn't find out otherwise. We intend to do that with regard to the war gaming protocols as well. The process that is ongoing right now is one whereby an elite group at Mitre is putting together a straw-man. We are assembling participants in a relatively small group, probably under ten people, from key organizations that will be involved with the war games. They will come up with a sort of a level-one protocol to let the war games interoperate. As that process proceeds and we get firmer, that will be, if you will, an equivalent to the current SIMNET protocols, we would expect to transition this activity into a more formal standards arena. But we find that you have to have this process where first you throw up a straw-man; you try it out, you do some prototyping. The presentation -- I borrowed one slide from here that Jim Wargo gave earlier -- talked about a test bed. You really have to have a real world environment to try these things out in. You have to, in the words of Fred Brooks, "build one to throw away". You have to do that first in order to be ready to build the right thing later. So we are going to go through that process, and we believe that when we are done we will be ready to make the war games interoperate. Then the exciting point comes when you tackle the harder research issue of how you make them interoperate with the other levels of simulation. It's not obvious how we are going to do that, but we believe we can do it. We believe we have the key ideas. So, we're excited to go forward with that. I've used up my two minutes plus one more probably; so why don't we go on to the BFIT tape because we're also very excited about the results of that activity.

(BFIT tape presentation made here)

WORKING GROUP SUMMARIES

Terrain Database Working Group Summary, Dexter Fletcher, IDA

We spent a number of minutes discussing things that we felt we should do for ourselves. It's very clear that we're next in the standards process. The communication sub-committee, of course, has the main lead right now with the specific suggestions and specific issues that must be hashed out, but it's coming in our direction next. What is also clear is that our sub-groups need to be a little more focused as the subgroups are beginning to dig into the technical issues and there are some very, very substantial technical issues that we need to address. That's very clear. That's something that needs to be done soon.

We received a number of issues for discussion as a result of the 13 position papers, some of which were presented on Tuesday. I have one page of these issues here. I'll address these issues backwards.

Issue number one, or rather number four, as it turned out, was the desire to establish a Dynamic Terrain PDU. We agreed that would be most desirable. We thought that if it's not something we could do now, but it needed to be addressed by one of the sub-groups. So our sub-group will undertake that as a task.

The next issue was the oceanographic information. We felt that that is certainly important while serving a correlation or terrain data base group, although I guess the ocean isn't exactly terrain. The other feeling was that we should turn to the Navy for help, so we plan to very soon now contact the Navy Oceanographic and Atmospheric Center and see what they have to offer. That gets a little bit into security issues. I'm not sure how that will affect the issues but hopefully we will have some topics on that issue.

The next two had to do with terrain data bases. I'm sure the information concerning dynamic terrain and cultural features entities will be included in the data base information. If we develop a Dynamic Terrain PDU, we feel that that will address the issue as to the how to return the two.

In relation to Terrain Database identifiers, we aren't sure exactly what the context for this issue was. It's clear that there need to be labels and there needs to be a data directory, but I've got a comment to make about that in a minute, and I will. But first, let me say that this is one that we weren't quite sure exactly what we were expected to do. So much of that is covered by folks like you in the 2851 ETL's, the data base directory is all the information that we have. But, number one, it raises a very interesting series of issues.

This has to do with what kinds of data base should be required for DIS. The thing is that it's an interactive issue. You build up an example and if somebody tried to use it and they say, "Gee, I found this to be very handy and I found that to be absolutely inadequate and unacceptable. Wouldn't you go and try to fix it up?" The conversation starts. Sooner or later, out of that conversation interaction emerges. It's something that everybody would like to see. Now, we've got a start on the kind of data base information that we think is correct. We think we built it. As a matter of fact, we might have two starts on it. The issue is that to get it used, it's got to be used by the folks that eventually are going to use it. So this gave rise to a theme that seems to be common not only to our group but to other groups as well. Which is, what is the turning context? What is the requirement context for this thing that we're building? We really need to hear from the potential user, the potential user being the services. First, how do they plan to use this tool and, secondly, would they please try out what we've done so far so that we can begin this

conversation. So that's the response that we have to the four issues.

Beyond that, we discussed a little bit about the relationship between the interchange specifications as well as they now exist in 2851 and the Simulated Database Interchange Specification. The feeling was that there should be a common librarian for both of those and that folks should try them out. They should be available from a single source and we elected George Lukes again, since he was not there.

We only had one very firm recommendation for the whole group and that's probably not on a major issue, but it concerns the label bindings as a field of bits. It came up a number of times by specific association of 32 bits. It was associated with specific country names and so the names are bound to specific bits. We felt that that was information that deserved to be somewhere in a data base dictionary and not in the specification. That was the specification that came out. There may be similar pieces of specificity in finding it in the specification as it now exists. That was something that the folks felt strongly about and probably needs to be reviewed to make sure that there are not specific fields that might well be removed.

To return just briefly to this business of interchange formats, we decided that if we were going to decide the relative merits of the 2851 and SDIS interchange specifications, then we better set up some criteria for that. There are some folks who will be meeting in the next couple of days, to set up the criteria for examining the two interchange formats. Beyond that, we also had a recommendation to ourselves. We talked about establishing a steering committee for ourselves so that we would know what the agenda and the issues and topics were that would be appropriate for a meeting next time. That's it from the terrain data base working group. Thank you.

Communications Protocols Working Group Summary, Joe Brann, IBM

Actually, I'm only going to reference the material for the Interface and Time/Mission Critical Subgroups and Al Kerecman will cover the Communication Architecture Subgroup. Yesterday, you heard people talk about sub or sub-sub-groups. In the working sessions that we've had between the workshops (we had an additional meeting in March and a meeting in July) we established some lower-tier working groups of our own group; Radiation Critical events, Dead Reckoning algorithms, and Articulated Parts. Here are the names of the coordinator or contact point. I ask you to do two things. If you're interested yourself and weren't in this room over here today, feel free to contact those names and join the working groups. Secondly, if there's somebody in your own company whom you know has experienced knowledge in those areas, please use that name list to allow those people to contact them. The broader we make these groups without making them too large, the more we're going to get the interactions that we need on the subject areas and the expertise. We can already see progress in the meetings that we've had in January, March, July and August because we're getting input, we're getting Navy input. This thing is growing properly because we have interaction. This is a good way to keep it going.

We started out this morning with a tremendously long list. We had a four page list of recommendations. Let me go over them one by one quickly.

Orientation should be represented using Euler angles. That was a voted-upon decision following up the recommendation in yesterday's position paper. (Some of these subject areas came out of the position papers yesterday. Some of them have come out of prior working meetings that we brought forth to this meeting in a larger context to get a larger reverification of it).

Angles should be represented in BAMS, Binary Angular Measurement, with orientation being thirty-two bit BAMS and articulated parts, whether it be a radar on a ship, a tank or whatever, represented in sixteen-bit BAMS. The definitions of what the sixteen and thirty-two bit BAMS are will be specified in the standard as well.

The Appearance PDU will have an additional field appended to it. I don't think we got a name for the field. That will be a sort of dead reckoning class, or a dead reckoning type, so that the sender can communicate back and forth with the receiver as far as what level, or order, or dead reckoning they're using. I understand that the contract between two simulators as far as being able to keep the object in check is that you must be using at least the same dead reckoning order as the sender is when he's using his dead reckoning to figure out when he should transmit the new Appearance PDU.

All we did today was define the fact that there will be a field that allows the IST staff to structure the PDU. The contents will be followed up by a sub-group headed by Dr. Jerry Burchfiel from BBN, which will delineate the algorithms, the equations and the numerations that will be used in that field.

Simulation management functions should establish the default update thresholds and establish a minimum default update rate. Working group in March decided from input that we should not use static threshold values to determine when my dead reckoning model separates from my real model (that's the trigger point for sending out the Appearance PDU). Motions were worked on in March to have the dynamic error thresholds in the standard. That is called the Update PDU. What we discussed today was being able to not only adjust that on a simulator interaction but to have a mechanism such that a battlemaster or a master control, a test operator can say, "Okay, right now all air specialists are going to be this."

They're going to be doing studies as part of the management part. The same thing for the default update value. How often do I have to put out an Appearance PDU if I'm either standing still or going in straightforward linear motions or will dead reckoning keep me going straight for a fixed amount of time? That will be handled as a management function and cannot be determined at this time. A PDU that will be associated with the management is the Activate PDU. We heard in Chris Pinon's presentation yesterday that management is still one of those areas that still needs work.

Articulated Parts representation is used to represent orientation of articulated parts. There is a sub-group that will be addressing that. Richard Schaffer is the head of that subgroup.

Detonation PDU. When a Detonation PDU has been issued, it will contain the coordinates of the target, the coordinates of the detonation location, both the energy and the directionality of the impacting weapon, the result of the detonation, and the range in meters. There is still discussion going on between the structure of the Fire and the structure of the Detonation PDU. This is what was decided should be included in the Detonation PDU. The thought here was that I can tell you where I hit you. You can figure it out if you want to. You can ignore my indication of where I hit you, use the coordinates on the other one to figure out where you may have been hit. But having the entity coordinates sent by the shooter will maybe assist the low fidelity trainer that doesn't want to go through all the population, maybe some backward dead reckoning to figure out where the guy was at the time of the detonation. This sort of covers both areas.

For the definition of what is indirect and what is direct fire, we will use ideally the service's definitions on that.

Query PDU's, which was brought up yesterday by Randy Saunders and Mike Robkins from Hughes, will be examined later on. What we

need right now is more closure on it, some sample structures to review and evaluate.

Time stamps: This topic was started in March and revisited in the July meeting and just reaffirmed at this meeting. They will be 32 bit unsigned integers. What that will do is divide an hour by 2^{31} parts and, when necessary, there will be a mechanism to specify the time stamp as relative to the hour or absolute.

You heard Alan Oatman yesterday talk about the ACME Radar PDU. We decided, based upon his needs and upon the fact that there isn't anything else suggested that supports his needs right now, that we will operate with the ACME Radar PDU for emissions. Actually, the PDU that you saw yesterday has been changed. The sub-group is now working actively to develop a final PDU for the standard. So there are changes. When that thing finalizes shortly, they will pass on the information to the IST project staff. At least in the interim, the result will be something close to the ACME Radar PDU you saw yesterday.

Byte ordering will be specified using diagrams to solve the "Big Endian - Little Endian" problem from a long time ago. Just for clarification, the completeness of that position in relation to how the bits within the bytes will be ordered and numbered will be presented in diagram.

Muzzle flashes: We can handle all of that in the Fire PDU. The bits in the Appearance PDU will be removed, because the Fire PDU was the best place to keep muzzle flash information.

Angular rotation rates: These will be represented by thirty-two bit signed integers representing BAMs per millisecond. Somebody very carefully brought up the fact that the current standard in BAMs per second only accommodates about half a revolution per second. This one probably already does it, but at

least it will accommodate everything we know so far in high rotation rate.

Last one, World coordinates: Three elements each, sixty-four bit floating point numbers. Entity coordinates, three elements, thirty-two bit floating point numbers.

That's a quick summary of the major actions, of major decisions by the Interface and Mission critical group. I'll be glad to entertain any questions that I can or focus you to the proper point of contact. Thank you very much.

Communication Architecture Sub-group, Al Kerecman, USACECOM

The communication architecture sub-group had an agenda which I think we made a good attempt at trying to cover. We had the following presentations:

1) A review from the March meeting
2) An update from UCF/IST
3) A very short presentation on the what's going on to incorporate voice networks, simulators into existing capabilities
4) A presentation on BFIT

We had issues to address from position papers which you may not have addressed specifically. What we wanted to do and what we wanted to achieve is to get into the required network services by DIS. We felt that was important to put forth so that UCF IST could articulate those in the December standards.

Just to give you an idea of where we were coming from with regard to that, we want to make it absolutely clear what we're talking about. We are talking about DIS services that are required by the layers below. This is a snapshot of the direction that things are going with regard to the effort of UCF/IST. So what we're going to do for you now is present to you what the committee decided on as the kind of services that were needed to support DIS. Please excuse the long acronym up top. What we're not addressing are those services that are required by other entities, other applicational entities that either can be serviced by existing protocol profiles or serviced by additions to the DIS PDU's, for example, handling command and control type information.

We don't have a view graph on this so let me read these to you and I'll try to articulate them to you as best I can. There's need for multi-cast, and broadcast capability, obviously. We felt, also, there's a need for point-to- point capabilities to handle the DIS specifications. We needed real-time delivery and we put an

upper bound on that real time delivery. Delivery must be less than five hundred milliseconds buffer to buffer. Now, of course, that varies with each PDU. It would have to be specified as a function of PDU's.

We're asking that package delivery be done in sequence as required. We believe that there are times when you want that to happen, where network services are more easily done for you. Perhaps it can deliver things quicker if they're done in sequence.

We believe that the specification must address the minimum delay dispersion. In other words, the packets must have a very similar delay tactic associated with it; they cannot have a variance.

There should be a minimum packet loss rate and we did not specify times or loss rate numbers here. We just were not ready to attach numbers. We think the fact that we would put in the concept that there's a minimal packet loss rate, the fact that we should draw your attention implies that parameter is worthy of your consideration.

One of things in the standard that needs to addressed is the idea that this can begin with something like a single local area network and go as far and wide as a Global networking of local area networks, globally interconnected using wide area network techniques.

The service needs both classified and unclassified capabilities with authentication when used in classified exercises. That authentication is really a driver for point to point; controlled access in communities of interest that are able to be partitioned and departmentalized.

The standard needs to use the ISO/CCITT guidance for site address assignments. We need a site administrator who assigns the specific inter-site or intra-site addresses and the exercise/test coordinator assigning addresses for members participating in the exercise. We need to set up a schema to identify specific multicast groups.

Finally, we need a transaction protocol to support the simulation management functions. The previous group also articulated the same voice.

The University of Central Florida's IST will now take these and put them into the appropriate form. I think that we probably will have another sub-group meeting before the December deadline so that we can make sure this is all put together properly. At that point these things should be incorporated into your standard to support DIS. Are there any questions? Good enough.

Performance Measures Working Group Summary, Bruce McDonald

There are certain things that happen inside the vehicle such as someone that's using passive sensors. When they go into a new mode, they don't tell the outside world about it. So, there's certain things that you need to know that are inside, and this is a way to get at that. So, what we propose is a performance measures request PDU. It will say, "For this measure go to table so-and-so, and I want you to do this in a certain phase." And, so, we're going to have to have a signal coming out to tell that simulator, "You're now in the target acquisition phase or attack phase, so go to this new set of measures that I want". There would be a header that would tell the simulator whether I want this transmitted during the exercise or stored and I'll ask for it later. Then we would have the entity addresses, "Who are these people I want this out of?" And then you would say, "Here are the number of measures that I'm going to ask for" and then you would say, "I want measure twenty-seven at rate so-and-so, let's say 15 Hz, I want measure 35 at once every five seconds." So the idea is that for this information, the simulator would go to its table and it would say, "He wants measures one, three and five. In this table, measure one is radar altimeter reading. This would be a way of communicating with a device that you need the information either during the exercise or you can ask for it afterwards. Now, we're not saying that we're telling all of these other simulators that don't have that capability that they have to create it. We're just saying if that simulator has this capability (and the higher end simulators certainly will), then we will be asking for that information and this is the way we would do it.

We call this a generic performance measures PDU. This is the way the entity or the simulator responds back to that command. It will have a header, naturally, there will be a time stamp that says, "This data was good at this point in time". And then it will say, "Here are the number of measures that I'm going to send you in

this PDU and here's the measure variable or the measure number that you asked for and here is what its value was at that time." You asked for three. "Here they are and here are their values." So this is a way of getting that data either during the exercise or afterwards.

There are certain times when an instructor or and evaluator just wants to know what is the condition in that simulator and so we're saying that you have a series of pre-defined variables that you want to know about. Say the evaluator has a menu and he selects an option to say, "Tell me what variable 27 is right now." This message goes out to the simulator and it's going to say, "It's coming from evaluator number two and sent to these people. I'm asking for this number of measures and here are the values I want back. And when you get this PDU, I want this information back."

Another thing is what we call an observed event PDU. Normally, if data is going into a data logger, we've got this mountain of data in there and a lot of times it's hard to figure out why all of a sudden this guy carried out this maneuver. Well, the idea of the observed event PDU is that an instructor looks for certain critical points, such as he just came over the horizon. Or he just got to the top of the hill. And he will enter a value that goes into the data logger that says, "So-and-so event has occurred" and so this data is going to make more sense. So that's what that is for.

And then we got into appearances. In the emitter PDU, one of the areas is Infrared. And so we feel that the type of information we need to know from this entity as it is moving is, "Tell me the temperature of the vehicle's skin, of the engine compartment, and the exhaust coming out." With that information, then the other entities can model what that would look like on infrared sensors from whatever angle. Because in the entity appearance PDU, we know where you are and what your attitude is. So with that information

and this infrared information, we figure you can then model what that guy will look like on your infrared display.

One of the white papers was on a hierarchial entity description. And our committee strongly agrees with that approach. We do feel that you do need to have your entities in a hierarchial structure and so, that's a recommendation of the Performance Measures Subgroup.

In the rationale document, we wrote up a description saying that you could define, say, five levels of obscuration (haze and smoke) and you could define it in terms of "I could see a target at a certain distance". The group decided that that is too limiting. That is very oriented toward a visual display when you've got all this infrared and emitting type information to calculate. So we're recommending that those five levels be defined in terms of atmospheric parameters such as humidity, temperature, and those kinds of things and then it's up to the manufacturer to figure out how far away you can see that object under those conditions. Because those conditions also effect your electronic detection ranges.

The articulated part. You've got a ship out there with antennas rotating. Is that required? The criterion we chose for making this decision was, would it effect the behavior of anybody in another simulator? And our decision is no. You're not going to do anything different whether that antenna is rotating or not, so why do it. We don't think you need a rotating antenna on a ship. Now, in the electronic world, it is very critical that my simulator know exactly where in the sweep you are. So, in the Emitter PDU, we do need a sweep position indication. Your data base will tell you at what rate that thing rotates. So we need a time hack that says, "I'm pointing north at this exact moment. Now, you dead reckon me for the next five seconds and you and I are both going to agree on where that thing is pointing. And after five seconds,

I'll tell you again where I am." So, the idea is electronically, we do need the antenna rotating and that visually, we don't see any value in it. You never get close enough to the ship to see the antenna rotating if it is an enemy ship, and if it is a friendly ship, you don't care anyway. In the other area of visual five-inch gun mount positions, we don't see a any value in representing that. What would you do differently if they moved? So, we don't see any reason to depict rotation of five inch gun mounts.

Ailerons and rudder positions. I was told by the fighter pilots at IST that you have to depict that. We had a long discussion on that. If you're teaching formation flight, then you need that, maybe. But DIS is not meant for initial skills acquisition, it's meant for going into battle together, and you would not use that information. If you're going after a bogey, you are going to shoot long before you can see its ailerons anyway. So, we don't think there's any value in transmitting rudder and aileron position to the rest of the world. Now, speed brakes, is a different matter. When a pilot pops the speed brakes you can definitely see that and it will effect your behavior. You can see that from a distance and it should be transmitted.

Regarding submarines. The first decision that was made was that parts on a submarine are not vernier, they are always either up or down. So it's not a true articulating part. So in a submarine, we only need up and down for the periscope, the snorkel, the radar, the missile launcher, and the ESM loop. So those are the things we need to represent coming up and down. You don't need an intermediate position. A snorkel, when it is in use, there will be smoke coming out. And so we would need to represent smoke coming out if that ship has decided to run the engine.

So those are the decision we came to in the sub-group. Any questions?

(Question) What do you do when you do not have a value that must be transmitted in a PDU?

You're going to have to transmit that field. You can't leave it out. It may be 999 or something. But you've got to transmit that field. Even if you don't have a value for it. You can't just leave out the field and chuck the PDU.

(Question) The DIS you describe seems to be designed for low fidelity simulators. Are we going to include high fidelity simulators in other types of problems?

I want to make a differentiation. You're saying that the fact that we don't need to see the ailerons move means we don't need the high fidelity simulators in the DIS network. I'm not saying that. I'm saying I don't see any need for that fidelity. The Performance Measures Subgroup doesn't see where it would modify the participants' behavior and if you're determined to put it in there to impress an Admiral, fine. Go ahead and put it in there. But right now I don't see any reason for transmitting that from one entity to another.

(Question) I don't think that was quite the question. I think there was quite a bit of discussion about what is the intended scope for the specification and how we think that needs to be further refined. Because there was a lot a discussion about, well, you could drop x out if you knew what the training requirements were for some specific kind of task. But the concern that we have for reading the scope in the rationale document and from the fact that nowhere in the name of the specification does, for instance, the word "training" or "tactical" appear, is that there are people who are going to want to take and apply this standard to all simulators that the government procures and in particular apply them to engineering simulators so that those simulators can interoperate with manned systems for evaluation of new weapon

systems as they're being designed. And that might be a very wonderful thing. I personally think that would be a very wonderful thing. But if you are going to include those kinds of things in the scope of the standard, you need to say you're going to include those kinds of things in the scope of the standard. We need a mechanism for negotiating overhead that's going to allow all of the features that are needed by all of the guys who are doing some kind of very accurate engineering simulation to be incorporated into this standard and that's something that the people in the time admission critical group could accept either way as long as it got written down and we were all agreed that that was what the standard was going to be used for.

(Question) I was also in that group and I agree with Dr. McDonald. No matter how faithful your simulation is within a model. I mean, I may have a very high fidelity aircraft and I care where my ailerons are, but I don't consider that within my model, I have to conform to the standards. My aerodynamics talking to my flight control doesn't send a PDU. However, from one model to another, yes, I care. But in that sense, I think Dr. McDonald's distinction applies. Nobody is going to do something different because I do something to my ailerons.

(Question) How can you make this limitation now based on where we are in developing this standard? We may not see any use for them now, but I don't think we should artificially limit the standard at this time.

(Question) I'd like also to say that obviously there is an overlap between what your group discussed and what her group, the communication protocol, discussed, and I'd like to convey what I think we agreed to when we discussed this same issue. And I think we agreed that we're not going to require everybody to put out articulated parts and we're not going to forbid them from doing it. All that we were going to do was, if they want to put it out, we

would tell them how. And then in this way we'll be flexible enough to accommodate applications that require it and don't require it.

The recommendation is that we need a standard appearance PDU followed by an articulation PDU. Maybe you can send one out right after the other. The high fidelity guys read both of them and the low fidelity guys ignore that second one. That's a possibility.

(Question) Let me raise an issue that appears to be one of fidelity, but it's really not. I happen to like the articulated parts for the submarine. However, I would ask the question, was consideration given to a submarine operating against a submarine or two submarines?

When a submarine is against a submarine, they don't see each other.

No, but they have sensors that can detect each other.

Yes, and that's in the acoustical data which is transmitted in the Emitter PDU. This applies to the visual data in the appearance PDU. So, yes, we address that.

Okay. Thank you.

(LtCol Sarner) Yesterday we had a presentation on replacing the bit encoding with these text strings that describes the entity. Now, I don't recall a recommendation from the protocol working group on that. Did you guys address that today?

(Al Kerecman) Yes we did. What we did indicate was we will pass it back to the project staff to put forth some examples. We do want to have the considerations of the difference between dynamic information and static information and coupled with that will be

the decision about going with bit coding or whether they go with the strings hierarchy of that group.

(LtCol Sarner) This recommendation here and some of the other ones about articulated parts seem to predispose that we're still going to have the number of bits that encode the position of the ailerons or whatever. Is that true?

(Bruce McDonald) No, we're saying that it's not required for performance measures.

(LtCol Sarner) Okay, well, the position of the speed brake that you've got bits in.

(Bruce McDonald) Now that one we do think should be transmitted.

(Question) This may be a nit, but if you're not going to do rudders and ailerons what about wing sweep on a variable geometry aircraft?

(Bruce McDonald) We forgot about that and I think that could be seen from a pretty good distance. I would think that you're going to need that.

(Question) I think there's also a little bit of confusion on low fidelity, high fidelity. Everyone's assuming if you're high fidelity, you can produce all this stuff, but only if there's somebody out there high fidelity who cares to listen. So there's a little confusion about who do you know who you're talking to and just because you're high fidelity doesn't mean it's some other high fidelity listening.

(Question) There seems to be a lot of interest in articulated parts here. As the sub-group chairman for articulated parts and

time mission critical group, I'll put a sheet up in the front for people to come up who are interested in addressing this.

(Question) If we're finished with the articulated parts, if you could put up the I.R. slide again. As I looked through your parameters on the I.R., my concern there was in our experience in doing I.R. modeling that the parameters you provided are sufficient for detection but they're not sufficient for recognition. Therefore, I can't discriminate the difference between the F-14 and a MIG-29. They're an I.R. spectrum. So I don't know whether to shoot or not to shoot. If I had weapon-free environment, I shoot everything. But the limited parameters you've provided here, all I can tell is that an object exists.

(Bruce McDonald) Keep in mind that this is in a PDU that has a location orientation relative to you. You can calculate his angle off of your nose. You know exactly whether or not that source is masked and you can figure that out. You should have enough math model to figure that out.

(Question) The issue is not whether he's visible or not. The issue is the characteristics of the model. This goes back to the low fidelity, high fidelity issue again. In looking at I.R. signatures, particularly of helicopters, if you look at an SH-3 helicopter and compare its I.R. signature and characteristics with that of a Sea Sprite, they are quite different. Visually, they're obvious. In I.R., you don't have all that information to use, so you have to look for very subtle characteristics to determine the difference between the two helicopters.

(Bruce McDonald) But remember that your simulator who's observing that entity knows it's a MIG.

(Question) How?

(Bruce McDonald) Because the whole theory of DIS is that every simulator has ground truth. You start sending fake data to the simulators, you blow the whole concept.

(Question) I'm driving the point that if somebody's sitting over here flying a MIG as an opposing force and I'm sitting over here flying or as a ground attack or anti-air player in the scenario, I don't know that's a Mig coming at me.

(Bruce McDonald) Yes, you do. Your computer does. You don't.

(Question) My user doesn't.

(Bruce McDonald) No, but all that is of importance is that your simulator knows it a MIG. It's at this orientation, it knows how to model the imagery that now you show to the users. That simulator knows it's a MIG.

(Question) Now we get to the issue of the high fidelity, low fidelity game. Because if you only give me a skin temperature and you give me engine compartment temperature, you aren't giving me wing temperature which is air speed driven, you aren't giving me other characteristics. You aren't giving me electronic package peak characteristics, which are very distinctive also. There are characteristics of these different models in the I.R. spectrum which are very specific and very unique. So you can always recognize that something is coming, but you can't tell what it is. The trainee can't. My high fidelity simulator knows it's a Mig, but where do I put the I.R. characteristics if you don't feed me the I.R. data. The dynamic data is I.R. characteristics.

(Bruce McDonald) I'd like a position paper on that.

(Question) We weren't done with articulated parts. He was faster with the microphone. I'm in the anti-articulated parts camp, and

I want to clear up one thing before we depart. We think you invented the greatest thing since sliced bread. Wing sweep on an airplane is nothing magic. We can get by with an icon that represents a MIG-23 or an F-14 or an F-111 because I defy you at 26,000 feet to decide what the hell it is out there whether his wings are forward or aft. They all look alike. So we can get by with a fixed wing. You only get a representation if he swept, he's probably fast; if it's forward, he's probably slow. So that would be a clue. The fact that you forgot to address sweeping wings is not to your detriment. I just wanted to clear up that one point. There's lots of things to address in the issues. That's why you need some different colored suits than gray pinstripes working a lot of these issues.

Closing Remarks, Brian Goldiez

Just a couple of things. I want to remind everybody that we've gotten all of this information on tape but if you have opinions, we need position papers, too. And silence is deadly. Those position papers are due by the first of October. This is a moving train and we need to capture what your needs are. You need to come up on the line and tell us.

There are over ninety different companies represented here over the past two days. I think that speaks that there's a lot of people interested in what's going on here and I want to thank everybody for their contributions. The next meeting we're planning will be in March. I think you're going to see in the coming months some shift in emphasis to testing, of continued growth and performance measures, correlation is going to become, if I have anything to do about it, a bigger deal. The steering committee over the coming month or two, I can assure you, will address the issues that were brought up by Tom Hoog of the Air force. Those will not be forgotten, but we're going to need some help from the government people to find out what their needs are. Thank you.

APPENDIX A

BASIC CONCEPTS
OF DISTRIBUTED INTERACTIVE SIMULATION

BASIC CONCEPTS
OF DISTRIBUTED INTERACTIVE SIMULATION

The basic concepts of Distributed Interactive Simulation (DIS) are an extension of the Simulation Networking (SIMNET) program developed by the Defense Advanced Research Projects Agency (DARPA). The purpose of DIS is to allow dissimilar simulators distributed over a large geographical area to interact in a team environment for the purposes of training, equipment development or equipment evaluation. These simulators communicate over local area networks and wide area networks. The basic DIS concepts are:

- No central computer for event scheduling or conflict resolution
- Autonomous simulation nodes responsible for maintaining the state of one or more simulation entities
- There is a standard protocol for communicating "ground truth" data
- Receiving nodes are responsible for determining what is perceived
- Simulation nodes communicate only changes in their state
- Dead reckoning is used to reduce communications processing

The implications of each of these concepts as they apply to DIS are discussed separately below.

No Central Computer

Some war games have a central computer that maintains the world state and calculates the effects of each entity's actions on other entities and the environment. These computer systems must be sized with resources to handle the worst case load for a maximum number of simulated entities. DIS uses a distributed simulation approach in which the responsibility for simulating the state of each entity rests with separate simulation nodes. As new nodes are added to the network, each new node brings its own resources.

Autonomous Simulation Nodes

The DIS nodes are autonomous and generally responsible for maintaining the state of one entity. In some cases, a node will be responsible for maintaining the state of several semi-automated forces entities. As the user operates controls in the simulated or actual equipment, the host computer in that node is responsible for simulating the resulting actions of the entity using a high fidelity simulation model. That node is responsible for sending messages to others, as necessary to inform them of any observable actions. All nodes are responsible for interpreting and responding to messages from other nodes and maintaining a simple model of the status of each entity on the network. All nodes also maintain a model of the status of the world including bridges and buildings that may be intact or destroyed.

Ground Truth Versus Perception

Each entity communicates to all other entities its current status (location, orientation, velocity, active emitters, articulated parts position, etc.). The receiving entity's host computer is responsible for taking this ground truth and calculating whether that entity is visible by visual or electronic means. This perceived status of the other entity is then displayed to the user on the simulated displays.

Dead Reckoning

In order to limit communications, each host computer maintains a simple model of the status of all other entities (within a given range) on the network. Between updates, the host computer extrapolates the position and orientation of the other entity based on its last reported location, velocity and acceleration. Each entity also keeps a simple model of its own state. When the state of the high fidelity model of ownship differs by a given amount from the simple model, the host computer sends out an update message to update the status of all simple models of the sending entity. This dead reckoning approach allows a host computer to update its display of the status of other entities at its normal update rate (e.g. 5, 15, 30, 60 HZ) while receiving updates in status from the other entities at a rate (about 1 Hz) that will not overload the communications network.

APPENDIX B

ATTENDEES LIST

ATTENDEES LIST

Bill Adams	RAYTHEON
Colin Agostini	Tracor Applied Science, Inc.
Robert Alger	Ford Aerospace
Richard Armstrong	USA Proj Mgr for Trng Device
Brian Ashford	Naval Training Systems Center
Janice Bachkosky	Naval Air Systems Command
Kevin Backe	ETL
Kevin Baner	Navel Ocean Systems Center
Arthur Banman	ABA Inc.
Lew Baraska	Information Spectrum Inc.
Mario Barbacci	Software Eng. Inst. CMU
George Barcus	NAVAIR-PMA 205
Michael Battaglia	Image Data Corp.
E. Bearfoot	Directorate of Army Training
Malcolm Bell	Concurrent Computer Corp.
Charles Benton	Technology Systems, Inc.
Steve Bettner	GE/SCSD
Bill Bewley	R&D Associates, DARPA-Europe
Benjamin Blood	Science Applications, Inc.
Steve Blumenthal	BBN Systems & Technologies
Nicolas Bogdanoff	THOMSON-CSF
Alvin Boudreaux	Defense Training & Perf. Ctr
John Brabbs	US Army TACOM
Daniel Bradford	Lockheed Sanders
Arnold Bramson	General Electric Co.
J. Brann	IBM Corp.
Jim Breiding	NTSC Code 221
Lawrence Brown	NTSC
Jerry Burchfiel	BBN Communications Corp.
Charles Burdick	Independent Test & Analysis
J. Burns	Naval Training Systems Center
Billy Burnside	U.S. Army
J. Cadiz	IST
Frank Calzaretta	BBN Systems and Tech. Corp.
Phyllis Campbell	Naval Air Systems Command
Brian Carlsson	NTSC
Mark Chaney	Developmental Simnet
Paul Chatelier	Perceptronics
George Chiesa	CAE Link Flight Simulation
Peter Chui	Hughes
Joel Clarion	Sogitec Electronique
John Collins	PM TRADE
Douglas Cook	Hughes Aircraft/Washington
Neale Cosby	IDA
Claude Crassous de Medev	Thomson CSF Division Simulation
Richard Crouse	NTSC
Larry Cruse	Naval Training Systems Center
David Curry	Armstrong A. Research Lab
Dean Curtiss	Allied-Bendix Test Systems Div.
Anthony DalSasso	Air Force
Jim Delle	McDonnell Douglas

Earle Denton	EL Denton Assoc.
J. Derema	Super Comp. Center of America
Mac McDiarmid	SAIC
Lance Diaz	Pathfinders Systems
Tom Dillon	Encore Computer Corp.
Gary Domaleski	Syscon Corp.
Kenneth Donovan	General Electric Company
Paul Donavon	Pathfinders Systems
Ken Doris	Grumman Aerospace
Dennis Doubleday	Software Eng. Institute, CMU
Carl Driskell	PM TRADE, Army
Paul Dumanoir	PM TRADE, Army
Woody Dunlap	AAI Corp.
Mark Easter	NATC
Edward Eberl	Amherst Systems, Inc.
John Eisenhardt	FAAC Inc.
Mark Eliot	Systems Development Division
Paul Ericksen	Cincinnati Electronics Corp.
Don Ethington	Worldwide Software Support Div.
James Evans	U.S. Air Force
Bahram Fatemi	FMC
Jean-Pierre Faye	Computer and Simulation Group
Robert Feeman	Air Force HQ FTD/SDR
Steve Fleming	Secretary of Air Force / AGPT
J. Fletcher	Institute for Defense Analysis
Stephen Food	Carnegie Mellon University
Tom Fortunato	Hughes
Kevin French	US Army TRADOC System Manage
Gary Freshour	AAI Corp.
Richard Gagan	Raytheon Co.
E. John Gagnon	Technology Systems, Inc.
Charles Gainer	Aviation R&D, ARI PERI-IR
Alan Galbavy	USA Proj Mgr for Trng Device
Dean Gammell	Planning Research Corp.
Dan Gareri	Planning Research Corp.
Dick Garvey	BBN
Thomas Gehl	IBM
David Geis	STI
Gary George	Link Flight Simulation Div.
Michael Georgiopoulos	IST
Jack Gifford	UNISYS
Robert Glasgow	IST
Lawrence Goldberg	Army CECOM AMSEL
Stephen Goldberg	U.S. Army Research Institute
Brian Goldiez	IST
Gilbert Gonzales	IST
Robert Goodwin	Commander Training Command LANT
Edward Gordon	Air Force ASD/ENETD
Raymond Green	Naval Training Systems Center
David Greschke	Simulator for Air to A. Comb, Luke AFB
Eric Gurd	US Army Tank-Automative Com. AMSTA-RVI
Thomas Hall	Simulation Concepts Corp.
Joe Haller	Grumman Corporation
James Hammond	NAVSEASYSCOM

Phil Handley	Perceptronics, Inc.
Mark Handrop	Air Force HQ-ASD 24/YIR
Doug Hardy	Naval Ocean Systems Center
Jim Harp	Silicon Graphics
William Harris	Naval Training Systems Center
Edward Harvey	BMH Associates, Inc.
Eric Haseltine	Hughes Test & Training Systems Div.
Kenneth Hawes	PM TRADE
Mike Healy	ETA Technologies Inc.
Jane Herman	Perceptronics, Inc.
James Hines	Planning Research Corp.
Carl Hobson	General Dynamics
Bob Hoeft	Naval Ordinance Center
Ronald Hofer	PM TRADE
Robert Hogue	Raytheon Company
Gene Holcomb	Harris Computer Corp.
Thomas Holl	Simulation Concepts Corp.
Thomas Hoog	Air Force ASD/ENET
H. Hopwood	Boeing
Judi Hruska	FMC Corp.
Greg Hudas	USA TACOM
Eric Imbert	U.S. Army TSM-SIMNET
Allen Irwin	SAIC
J. Jackson	Ministry of Def. Army U.K.
Bob Jacobs	Illusion Eng. Inc.
Christopher Jehn	OSD
Ron Jensen	Comptex Res. Inc.
Larry Jobson	Information Spectrum Inc.
Roy Jonkers	GTE-Government Systems
Patrick Kallaus	DMAAC
John Kammerer	Naval Ocean Systems Center
Michael Kamrowski	Air Force Human Resources Lab
Gary Kamsickas	Boeing Aero & Electronics
Craig Kanarick	BBN
Suhail Kassis	Hughes Aircraft Company
Amnon Katz	McDonnell Douglas
Al Kerecman	US Army CECOM
Dean King	USAF Aero. Systems Div. ASD/YWB
Dave Klindt	Klindt's Consulting
Brian Kline	IST
Samuel Knight	Link Flight Simulation Corporation
Fred Kolb	McDonnell Douglas
Lee Kollmorgen	DARPA / SIMNET
Mark Krikorian	Cross Systems, Inc.
Jerrold Kronenfeld	TASC
Bob Kruse	Boeing
Del Kunert	Concurrent Computer Corporation
Norman Lane	SIMNET Office
Eric Lane	BBN Systems & Technologies
Curtis Lawson	GTE/SSD Systems Engineering
Richard Layfield	IVEX
Nat League	AAI Corporation
Dale Leavy	NTSC
Norman Lessard	Hughes

Harry Levey	NTSC
Kurt Lin	IST
Wayne Lindo	Northrop Aircraft
Curt Lisle	IST
Margaret Loper	IST
Pablo Lopez	Naval Ordinance Station
Peter Loser	Competence Center Informatik
Heing Lotz	Bundesamt Fur Wehrtechnik
Bryan Lowe	McDonnell Douglas Elect. Corp.
Dan Lucero	Intelisys
Rich Macy	HQ USAF/INTB
Joyce Madden	Naval Training Systems Center
Marshall Magruder	Hughes
Dereck Malin	Ministry of Defense Pro. Exe
Farid Mamaghani	BBN Systems & Technology
Louis Mangiafico	Naval Underwater Systems Center
Fred Mangol	Information Spectrum, Inc.
Edward Martin	Air Force ASD/ENETS
George May II	Defense Mapping Agency
Dennis McBride	DARPA
Michael McCluskey	Army Research Institute PERI-IO
Bruce McDonald	IST
Ed McDonnell	Concurrent Computer Corp.
Jim McDonough	Illusion Engineering, Inc.
David McKeeby	Analysis & Tech., Inc.
Lou Medin	IST
Louis Medley	Naval Air Systems Command
Robert Meidenbauer	Amherst Systems, Inc.
Larry Meliza	Army Research Institute
Randy Michelsen	Los Alamos
Duncan Miller	BBN Laboratories
Laurie Miller	Ball Systems Eng. Division
Russell Miller	FORCES Command
Kent Miltenderger	HQ AFEWC/EWS
Allen Mitchell	BBN - Advanced Simulation Div.
Ron Moore	Boeing Military Airplane
Edward Moore, Jr.	NTSC
Ken Morris	Northrop Corp.
James Morrison	SRS Technologies
Michael Moshell	IST/UCF Computer Science Dept.
John Mudlo	Martin Marietta
Ronald Munzer	Ball Systems Engineering Div.
Eugene Naccarato	Planning Research Corp.
Henry Nelson	IST Visual Systems Lab
Mike Newton	ETA Technologies, Inc.
Steven Norman	General Dynamics Ft. Worth Div.
Alan Oatman	BBN-STC
Kenii Oda	Planning Research Corporation
Barbara Osbon	General Dynamics
Ruey Ouyang	IST
Stephen Overstreet	USA Project Mgr for Training Devices
Charles Owens	Harris Comp. Sys. Div.
Judith Pafford	Hughes
William Parrish	Naval Training Systems Center

Robert Paschall	Electronic Warfare Assoc.
Amy Patz	NTSC
Ben Patz	NTSC
Ben Paz	NTSC
Robert Persons	Motorola, Inc.
Ken Peterson	NTSC
Mikel Petty	IST
Pascal Peyronnet	Thomson-CSF Sims. Div./BP 22
David Pfahler, Sr.	Hughes Aircraft Co.
Christina Pinon	IST
Eytan Pollak	GE Aerospace
Arthur Pope	BBN Laboratories
Marie Pope	CAE Link Flight
Dave Pratt	Naval Postgraduate School
Micheline Provost	IST
Jeffrey Pulcini	Concurrent Computer Corp.
William Pulice	TRW
J. Mark Pullen	DARPA/TTO
Rolando Rabines	BBN Systems & Tech.
Marie Raney	Hughes Flight Simulation
Bill Reese	NTSC
Bob Reinwald	TPDC
Ronald Rhoades	Army Aviation Programs
Kurt Rhodehamel	Encore Computer Corp.
Paula Ridolfi	USAF ASD/YWD
Bruce Riner	NTSC
Bill Roberts	McDonnell Douglas Training Systems
Bruce Robinson	RH and Company
Michael Robkin	Hughes Simulation Systems, Inc.
W. Rowan	NTSC
Steven Sarner	USAF ASD/YWB
Alex Satkowski	SOGITEC
Randy Saunders	Hughes Simulation Systems
Richard Schaffer	BBN
Richard Schantz	BBN
William Schelker	Air Force ASD/ENETD
Donald Schmaltz	Integrated Software Resource
Ludger Schumann	Competence Center Informatik
Robert Schwing	Link Flight Simulation Corp.
Steve Seidensticker	Logicon, Inc
David Shea	IST
Shen Shey	MIT Lincoln Lab
James Shiflett	ICAF
Thomas Sicilia	Defense Training & Per. D.C.
Michael Sieverding	ARINC Research Corp.
Frank Sigmund	Loral Systems Group
Michael Sivret	DIGITAL
Hank Small	EER Systems
Mike Smallwood	Image Data Corp.
Joshua Smith	BBN
Stephen Smyth	BBN - Advanced Simulation
Terry Snyder	Grumman Sim. Trainer Products
Karl Spuhl	McDonnell Douglas Corp.
Steve Stephens	AFHRL/OTA

Wayne Stimpson	General Research Corp.
Daniel Sullivan	BBN
Steven Swaine	McDonnell Douglas
William Szymanski	PM TRADE
Mario Talana	NTSC
Ronald Tarr	Training & Performance Data Center
Donnie Tedder	Simulation Animation Service of Florida
Gary Teper	Information Spectrum, Inc.
Bernard Thiebaut	Thomson-CSF
Ed Thomas	Army Engineer School ATSE-TOT
Jack Thompson	IST
Jack Thorpe	DARPA EUROPE
Tom Tiernan	Naval Ocean Systems Center
Steven Trencansky	Concurrent Computer Corp.
Darrell Turner	Martin Marietta ISG
Michael Uecker	USAF ASD/YWSA
Milton Unnam	CAE-LINK
Thomas Verscharen	NTSC
Jay Wackler	Concurrent Computer Corp.
John Wang	Naval Underwater Systems Center
Jim Wargo	DARPA ADST
Wayne Warren	U.S. Army Training Support Center
Gary Washam	Megatek Corp.
Jim Watson	Florida Sparta
Barry Webb	Lockheed Aeronautics Systems
Charles Weinstock	Software Eng. Institute
Glenn Weissinger	General Dynamics Corp.
Ralph Whitney	Motorola, Inc.
Craig Wickham	Encore Computer Corp.
Gene Wiehagen	PM TRADE-AMCPM-TND-EC
Ed William	NTSC
David Williams	USAF ASD/ENETD
Henry Williams	IST
Ronald Williams	Concurrent Computer Corp.
Ruth Willis	NTSC
David Wood	MITRE Corp.
Pam Woodard	NTSC
Harry Zywiol	USA TACOM

END